In [2]:

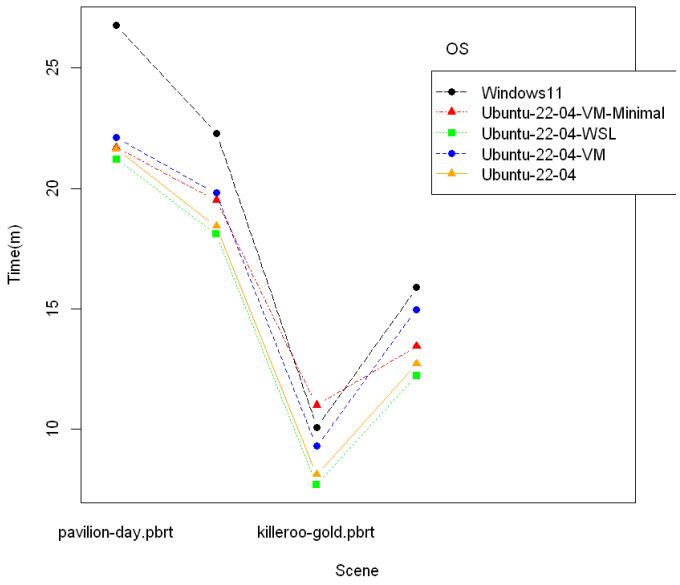
```r
# 1. Carga inicial de datos.
if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}

# Lectura de datos
Data <- read.csv2("updated_file.csv", sep = ",", header = TRUE)


Data$Time <- as.numeric(Data$Time)
Data$OS <- factor(Data$OS, levels = unique(Data$OS))
Data$Scene <- factor(Data$Scene, levels = unique(Data$Scene))
Data$Acc.Int <- factor(Data$Acc.Int, levels = unique(Data$Acc.Int))


# 2. Verificación de la lectura de datos.
library(psych)
headTail(Data)
str(Data)
summary(Data)


# 3. Gráfico simple de interacción.
# Variable dependiente: Time
# Variables independientes: OS y Scene.
interaction.plot(x.factor = Data$Scene,
 trace.factor = Data$OS,
 response = Data$Time,
 fun = mean,
 type = "b", col = c("black", "red", "green", "blue", "orange"),
 pch = c(19,17,15,19,17),
 fixed = TRUE, xlab= "Scene", ylab="Time(m)", trace.label="OS",
 leg.bty = "o")


# Variable dependiente: Time
# Variables independientes: OS y Acc.Int.
interaction.plot(x.factor = Data$Acc.Int,
 trace.factor = Data$OS,
 response = Data$Time,
 fun = mean,
 type = "b", col = c("black", "red", "green", "blue", "orange"),
 pch = c(19,17,15,19,17),
 fixed = TRUE, xlab= "Accelerator-Integrator", ylab="Time(m)", trace.label="OS",
 leg.bty = "o")

# 4. Modelo lineal
model <- lm(Time ~ Scene * OS * Acc.Int, data = Data)

# 5. Evaluación de los supuestos
x <- residuals(model)
library(rcompanion)
par(mfrow = c(2, 1))
plotNormalHistogram(x, xlab="Time(m)")
qqnorm(resid(model), main = "Normal Q-Q", xlab = "Theoretical Quantiles", ylab = "Standarized residuals")
qqline(resid(model), col = "red", lwd = 2)
par(mfrow = c(1, 1))
plot(fitted(model), residuals(model))
plot(model)
# La normalidad es debatible, sin embargo igual se puede apreciar.
# En el gráfico de dispersión no se observa un patrón claro entre los datos, del mismo modo se procede a rea
# prueba Levene para confirmar homocedasticidad.

leveneTest(Time ~ Scene * OS * Acc.Int, data = Data)
# La prueba Levene retorna un P-Value 0.9856, lo que sugiere homocedasticidad en los datos.

# Debido a que la normalidad es el supuesto más permisivo y que tanto el gráfico de dispersión y la prueba L
# sugieren la presencia de homocedasticidad en los datos, no se procede a realizar transformación de los mis
# Para no incurrir en el riesgo de acercar mucho los datos innecesariamente. (No incurrir en error tipo-II)

# 6. ANOVA
```

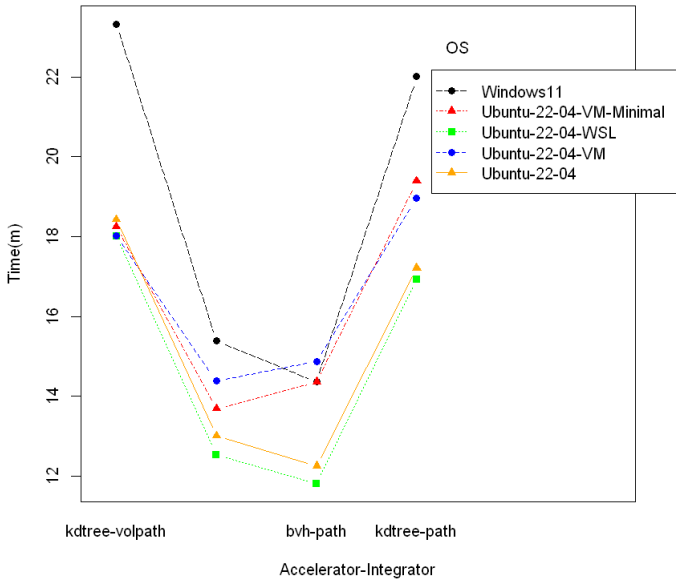
```r
library(car)
Anova(model, type = "II")

# 7. Gráfico bigotes con error estándar OS
Sum <- Summarize(Time ~ OS, data=Data, digits=3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits=3)
Sum

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=OS, y=mean, color = OS)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=OS), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0),
            legend.position="none") +
      ylab(expression("Time (m)")) +
      ggtitle("Time vs OS")

# 8. Gráfico bigotes con error estándar Scene:OS
Sum <- Summarize(Time ~ OS + Scene, data=Data, digits=3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits=3)
Sum

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Scene, y=mean, color = OS)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=OS), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Time (m)")) +
      ggtitle("Time vs Scene in function of OS")


# 9. Gráfico bigotes con error estándar Acc.Int:OS
Sum <- Summarize(Time ~ OS + Acc.Int, data=Data, digits=3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits=3)
Sum

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Acc.Int, y=mean, color = OS)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=OS), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Time (m)")) +
      ggtitle("Time vs Acc.Int in function of OS")

# 10. Pruebas T
# Prueba para OS
t_test_os <- pairwise.t.test(Data$Time, Data$OS, p.adjust.method = "BH")
# Prueba para Scene:OS
t_test_os_scene <- pairwise.t.test(Data$Time, Data$OS : Data$Scene, p.adjust.method = "BH")
# Prueba para Acc.Int
t_test_accint <- pairwise.t.test(Data$Time, Data$Acc.Int, p.adjust.method = "BH")
```

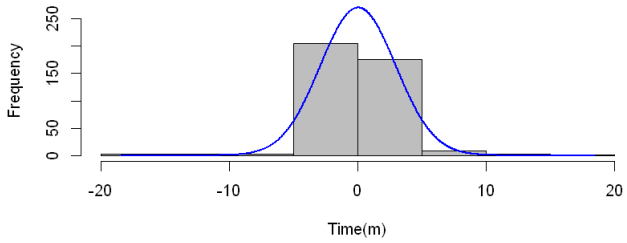
A data.frame: 9 × 4

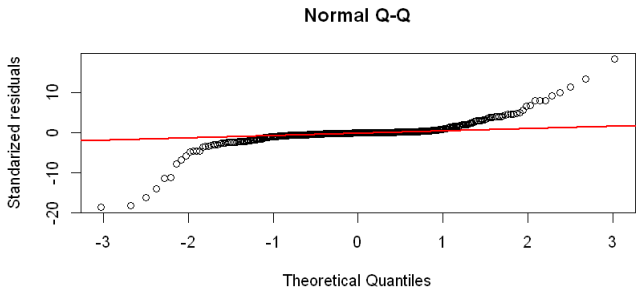
	OS	Scene	Time	Acc.Int
	<fct>	<fct>	<chr>	<fct>
1	Windows11	pavilion-day.pbrt	28.93	kdtree-volpath
2	Windows11	dragon-10.pbrt	33.35	kdtree-volpath
3	Windows11	killeroo-gold.pbrt	7.95	bvh-volpath
4	Windows11	dragon-10.pbrt	11.1	bvh-path
...	NA	NA	...	NA
397	Ubuntu-22-04-VM-Minimal	killeroo-gold.pbrt	6.64	bvh-path
398	Ubuntu-22-04-VM-Minimal	vw-van.pbrt	11.91	bvh-path
399	Ubuntu-22-04-VM-Minimal	killeroo-gold.pbrt	9.92	kdtree-volpath
400	Ubuntu-22-04-VM-Minimal	dragon-10.pbrt	11.48	bvh-path

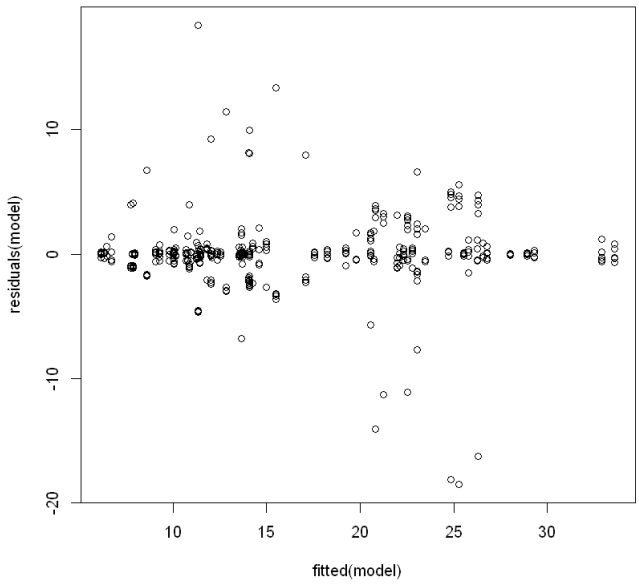
```
'data.frame':   400 obs. of  4 variables:
 $ OS     : Factor w/ 5 levels "Windows11","Ubuntu-22-04-VM-Minimal",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Scene  : Factor w/ 4 levels "pavilion-day.pbrt",..: 1 2 3 2 4 2 2 4 3 3 ...
 $ Time   : num  28.93 33.35 7.95 11.1 13.98 ...
 $ Acc.Int: Factor w/ 4 levels "kdtree-volpath",..: 1 1 2 3 2 2 4 4 3 1 ...
                       OS                    Scene          Time
 Windows11              :80   pavilion-day.pbrt :100   Min.   : 5.87
 Ubuntu-22-04-VM-Minimal:80   dragon-10.pbrt    :100   1st Qu.:10.17
 Ubuntu-22-04-WSL       :80   killeroo-gold.pbrt:100   Median :13.61
 Ubuntu-22-04-VM        :80   vw-van.pbrt       :100   Mean   :16.36
 Ubuntu-22-04           :80                            3rd Qu.:22.76
                                                       Max.   :34.43

          Acc.Int
 kdtree-volpath:100
 bvh-volpath   :100
 bvh-path      :100
 kdtree-path   :100
```

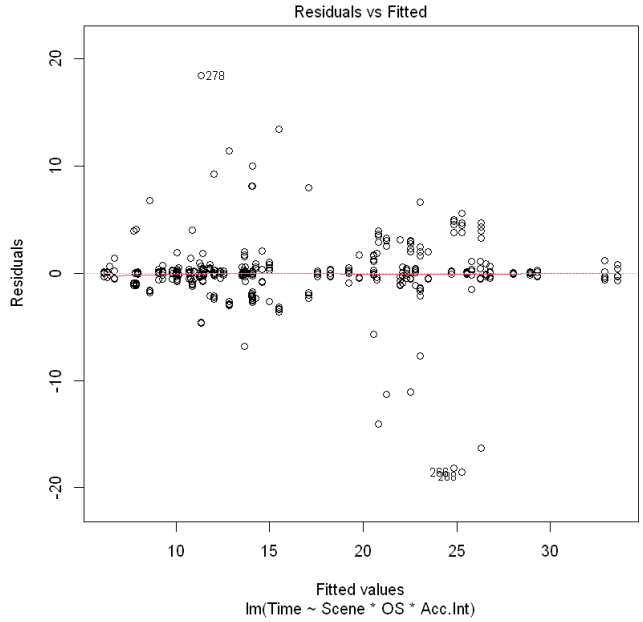


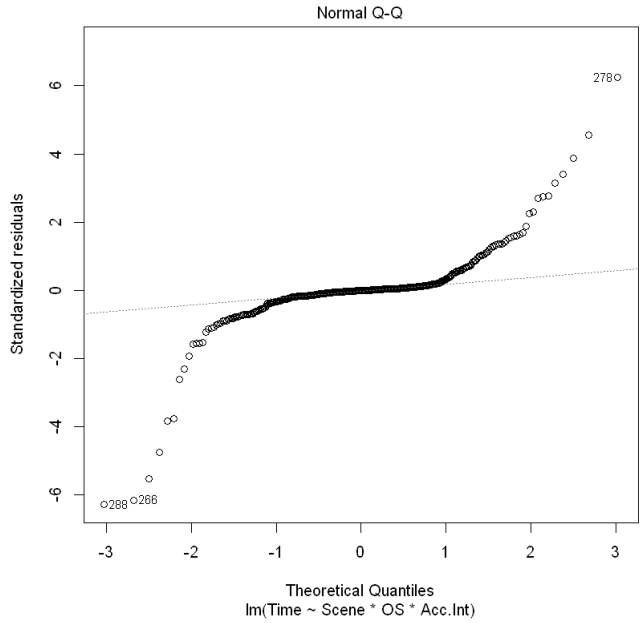

OS
Windows11
Ubuntu-22-04-VM-Minimal
Ubuntu-22-04-WSL
Ubuntu-22-04-VM
Ubuntu-22-04
Time(m)
12
14
16
18
20
22
kdtree-volpath
bvh-path
kdtree-path
Accelerator-Integrator

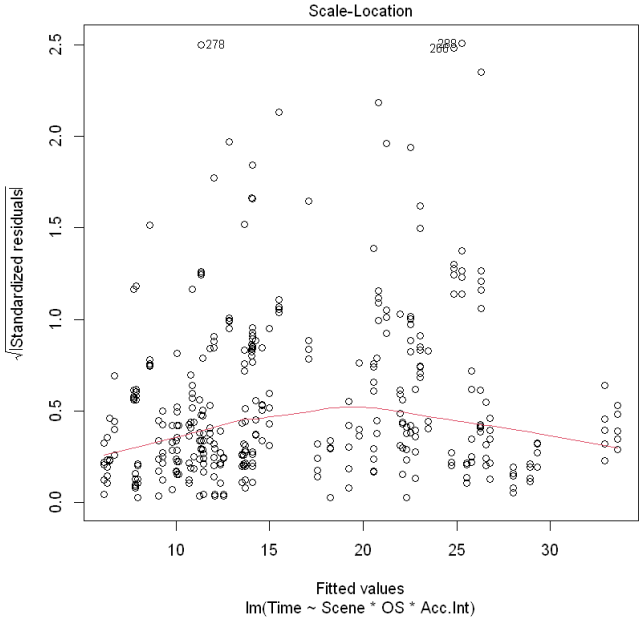

Frequency
0
50
150
250
-20
-10
0
10
20
Time(m)


Normal Q-Q
Standarized residuals
-20
-10
0
10
-3
-2
-1
0
1
2
3
Theoretical Quantiles


residuals(model)
-20
-10
0
10
10
15
20
25
30
fitted(model)


Residuals vs Fitted
Residuals
Fitted values
lm(Time ~ Scene * OS * Acc.Int)
278
266
288


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Time ~ Scene * OS * Acc.Int)
278
288
266


Scale-Location
√|Standardized residuals|
Fitted values
lm(Time ~ Scene * OS * Acc.Int)
278
288
266

A anova: 2 × 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	79	0.7474269	0.9396665
	320	NA	NA

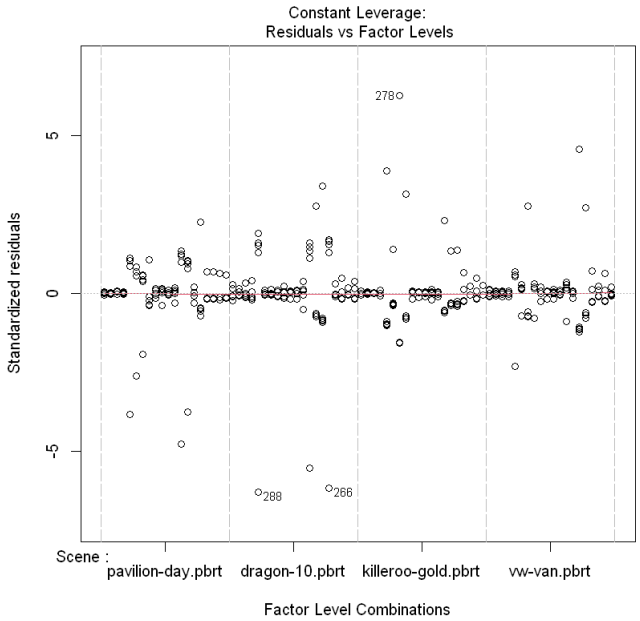
A anova: 8 × 4

	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Scene	10765.5178	3	330.7342776	1.094224e-97
OS	761.1460	4	17.5377359	5.018005e-13
Acc.Int	2914.9651	3	89.5524861	4.293080e-42
Scene:OS	224.0382	12	1.7207050	6.127817e-02
Scene:Acc.Int	4183.9836	9	42.8462679	6.492352e-50
OS:Acc.Int	255.1001	12	1.9592726	2.739847e-02
Scene:OS:Acc.Int	267.8452	36	0.6857201	9.152851e-01
Residuals	3472.0378	320	NA	NA

A data.frame: 5 × 10

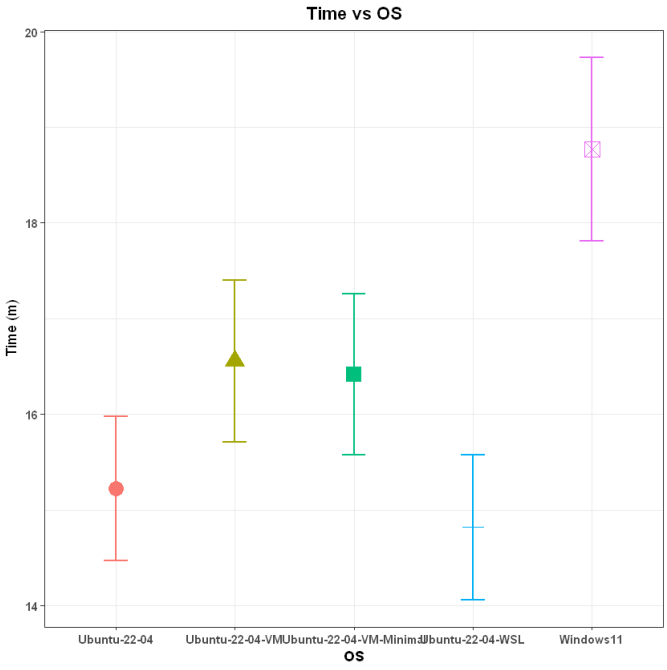
OS	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Windows11	80	18.770	8.601	7.77	12.010	15.795	26.230	34.43	0.962
Ubuntu-22-04-VM-Minimal	80	16.419	7.509	6.64	11.018	13.665	22.625	30.84	0.840
Ubuntu-22-04-WSL	80	14.819	6.798	5.87	9.480	12.025	21.028	27.39	0.760
Ubuntu-22-04-VM	80	16.555	7.566	6.64	11.108	13.670	22.933	31.00	0.846
Ubuntu-22-04	80	15.224	6.761	6.15	9.992	13.285	21.990	27.44	0.756

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead."
```



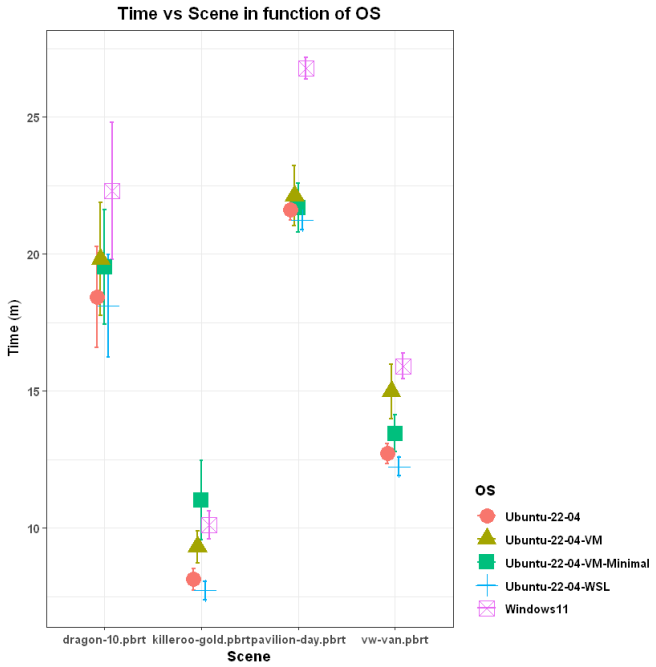
A data.frame: 20 × 11

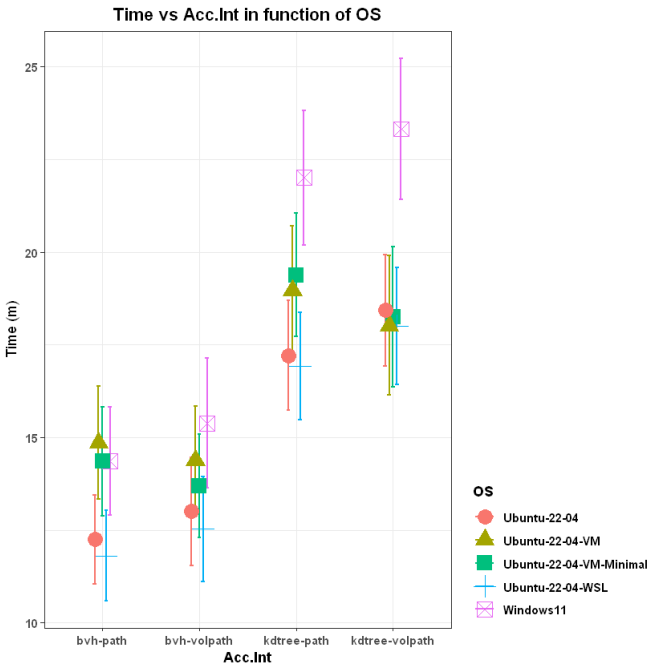
OS	Scene	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Windows11	pavilion-day.pbrt	20	26.784	1.774	24.57	25.285	26.790	28.258	29.00	0.397
Ubuntu-22-04-VM-Minimal	pavilion-day.pbrt	20	21.697	4.008	9.93	21.168	22.205	24.540	25.50	0.896
Ubuntu-22-04-WSL	pavilion-day.pbrt	20	21.220	1.515	18.32	20.235	21.235	22.613	23.23	0.339
Ubuntu-22-04-VM	pavilion-day.pbrt	20	22.126	4.979	6.79	21.548	22.650	24.778	29.66	1.110
Ubuntu-22-04	pavilion-day.pbrt	20	21.628	1.771	19.33	20.278	21.990	22.922	25.49	0.396
Windows11	dragon-10.pbrt	20	22.304	11.235	10.84	11.268	22.230	33.085	34.43	2.510
Ubuntu-22-04-VM-Minimal	dragon-10.pbrt	20	19.520	9.404	6.76	11.727	12.200	29.410	30.84	2.100
Ubuntu-22-04-WSL	dragon-10.pbrt	20	18.105	8.404	9.29	10.110	17.280	26.328	27.39	1.880
Ubuntu-22-04-VM	dragon-10.pbrt	20	19.814	9.232	6.71	11.732	17.150	29.593	31.00	2.060
Ubuntu-22-04	dragon-10.pbrt	20	18.430	8.210	9.97	10.280	18.970	26.300	27.44	1.840
Windows11	killeroo-gold.pbrt	20	10.094	2.278	7.77	7.878	9.940	12.328	12.73	0.509
Ubuntu-22-04-VM-Minimal	killeroo-gold.pbrt	20	11.008	6.434	6.64	6.787	9.670	9.990	29.74	1.440
Ubuntu-22-04-WSL	killeroo-gold.pbrt	20	7.713	1.532	5.87	6.315	7.485	9.270	9.50	0.343
Ubuntu-22-04-VM	killeroo-gold.pbrt	20	9.304	2.581	6.64	6.875	9.755	10.110	15.34	0.577
Ubuntu-22-04	killeroo-gold.pbrt	20	8.122	1.757	6.15	6.300	8.610	9.250	12.00	0.393
Windows11	vw-van.pbrt	20	15.898	2.095	13.39	13.898	15.795	17.785	18.59	0.468
Ubuntu-22-04-VM-Minimal	vw-van.pbrt	20	13.452	2.998	6.86	11.982	12.495	15.170	22.16	0.670
Ubuntu-22-04-WSL	vw-van.pbrt	20	12.237	1.488	10.15	11.008	12.025	13.660	14.25	0.333
Ubuntu-22-04-VM	vw-van.pbrt	20	14.977	4.418	11.91	12.328	13.670	15.322	28.91	0.988
Ubuntu-22-04	vw-van.pbrt	20	12.714	1.652	10.65	11.050	13.285	13.752	16.69	0.369



A data.frame: 20 × 11

OS	Acc.Int	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Windows11	kdtree-volpath	20	23.324	8.564	12.38	16.622	23.675	29.980	34.43	1.91
Ubuntu-22-04-VM-Minimal	kdtree-volpath	20	18.247	8.452	6.76	9.930	15.510	24.312	30.84	1.89
Ubuntu-22-04-WSL	kdtree-volpath	20	18.012	7.083	8.75	12.050	18.035	23.630	27.39	1.58
Ubuntu-22-04-VM	kdtree-volpath	20	18.021	8.388	6.79	10.150	15.635	24.573	31.00	1.88
Ubuntu-22-04	kdtree-volpath	20	18.429	6.715	9.25	13.312	19.340	24.942	27.44	1.50
Windows11	bvh-volpath	20	15.383	7.813	7.83	10.465	12.890	17.718	28.09	1.75
Ubuntu-22-04-VM-Minimal	bvh-volpath	20	13.690	6.243	6.74	11.318	12.120	13.330	25.50	1.40
Ubuntu-22-04-WSL	bvh-volpath	20	12.527	6.337	5.94	8.773	10.350	13.775	23.23	1.42
Ubuntu-22-04-VM	bvh-volpath	20	14.374	6.551	6.79	11.725	12.235	17.045	25.58	1.46
Ubuntu-22-04	bvh-volpath	20	13.008	6.531	6.29	9.432	10.990	14.890	25.49	1.46
Windows11	bvh-path	20	14.363	6.486	7.77	10.102	12.790	16.567	24.91	1.45
Ubuntu-22-04-VM-Minimal	bvh-path	20	14.354	6.596	6.64	11.367	11.925	21.695	29.74	1.47
Ubuntu-22-04-WSL	bvh-path	20	11.810	5.475	5.87	8.547	10.140	13.402	20.82	1.22
Ubuntu-22-04-VM	bvh-path	20	14.863	6.844	6.64	11.533	12.035	21.913	28.91	1.53
Ubuntu-22-04	bvh-path	20	12.246	5.388	6.15	9.512	10.645	14.967	22.57	1.20
Windows11	kdtree-path	20	22.009	8.109	11.80	16.030	21.575	27.320	34.09	1.81
Ubuntu-22-04-VM-Minimal	kdtree-path	20	19.386	7.443	9.59	13.992	20.880	26.058	29.60	1.66
Ubuntu-22-04-WSL	kdtree-path	20	16.927	6.434	8.50	12.420	16.120	20.870	26.91	1.44
Ubuntu-22-04-VM	kdtree-path	20	18.962	7.844	6.71	13.592	18.090	25.938	29.83	1.75
Ubuntu-22-04	kdtree-path	20	17.212	6.619	9.08	12.498	16.710	22.555	27.36	1.48




Time vs Acc.Int in function of OS
Time (m)
25
20
15
10
bvh-path
bvh-volpath
kdtree-path
kdtree-volpath
Acc.Int
OS
Ubuntu-22-04
Ubuntu-22-04-VM
Ubuntu-22-04-VM-Minimal
Ubuntu-22-04-WSL
Windows11

In [ ]:

In [1]:

```
# ANOVA Monofactorial

# 1. Carga inicial de datos:

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(Rmisc)){install.packages("Rmisc")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(multcompView)){install.packages("multcomp")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


Datos <- ("
Algoritmo       Ejecucion  Tiempo
'Algoritmo A'      '1'    12060
'Algoritmo A'      '2'    14089
'Algoritmo A'      '3'    13502
'Algoritmo A'      '4'     9574
'Algoritmo A'      '5'    14056
'Algoritmo A'      '6'    11569
'Algoritmo A'      '7'    13047
'Algoritmo A'      '8'    13275
'Algoritmo A'      '9'    14257
'Algoritmo A'      '10'   15075
'Algoritmo A'      '11'   12506
'Algoritmo A'      '12'   11557
'Algoritmo A'      '13'    9548
'Algoritmo A'      '14'   11514
'Algoritmo A'      '15'   16015
'Algoritmo A'      '16'   13004
'Algoritmo A'      '17'   10510
'Algoritmo A'      '18'   13040
'Algoritmo A'      '19'   17098
'Algoritmo A'      '20'   13080
'Algoritmo B'      '1'    11080
'Algoritmo B'      '2'    12089
'Algoritmo B'      '3'    12538
'Algoritmo B'      '4'    10571
'Algoritmo B'      '5'    12010
'Algoritmo B'      '6'    12598
'Algoritmo B'      '7'    13543
'Algoritmo B'      '8'    13547
'Algoritmo B'      '9'    13217
'Algoritmo B'      '10'   15297
'Algoritmo B'      '11'   12210
'Algoritmo B'      '12'   11299
'Algoritmo B'      '13'   10067
'Algoritmo B'      '14'   11279
'Algoritmo B'      '15'   14006
'Algoritmo B'      '16'   12099
'Algoritmo B'      '17'   11581
'Algoritmo B'      '18'   14012
'Algoritmo B'      '19'   15069
'Algoritmo B'      '20'   12000
'Algoritmo C'      '1'     9081
'Algoritmo C'      '2'    11012
'Algoritmo C'      '3'    11529
'Algoritmo C'      '4'     9569
'Algoritmo C'      '5'    11092
'Algoritmo C'      '6'    11524
'Algoritmo C'      '7'    12522
'Algoritmo C'      '8'    12588
'Algoritmo C'      '9'    12241
'Algoritmo C'      '10'   13257
'Algoritmo C'      '11'   11294
'Algoritmo C'      '12'   10226
'Algoritmo C'      '13'    9591
'Algoritmo C'      '14'    9224
'Algoritmo C'      '15'   12033
'Algoritmo C'      '16'   11063
'Algoritmo C'      '17'    9537
'Algoritmo C'      '18'   13014
```

```
'Algoritmo C'      '19'   14033
'Algoritmo C'      '20'   11093
")

# Lectura de los datos
Data <- read.table(textConnection(Datos), header=TRUE)
# Ordenar los datos segun los ingresamos
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))


# 2. Lectura de datos / Verificación de lectura

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(Datos)

# 3. Resumen organizado

Summarize(Tiempo ~ Algoritmo, data = Data, digits = 4)

# 4. Diagrama de cajas

M <- tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

# 5. Información de promedios e intervalos de confianza

Sum <- groupwiseMean(Tiempo ~ Algoritmo, data = Data, conf = 0.95, digits = 3, traditional = FALSE, percenti
Sum

# 6. Gráficos de promedios e intervalos de confianza

library(ggplot2)
ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                     geom_point(shape = 15,
                                size = 4) +
                     theme_bw() +
                     theme(axis.title = element_text(face = "bold")) +
                     ylab("Tiempo promedio, s")

# 7. Modelo Lineal

model <- lm(Tiempo ~ Algoritmo, data = Data)
summary(model)

# 8. ANOVA

library(car)
Anova(model, type = "II")

# 9. Histograma de residuos

X <- residuals(model)
library(rcompanion)
plotNormalHistogram(X)

# 10. Dispersión de residuos

plot(fitted(model),residuals(model))

# 11. Gráficos del modelo lineal

plot(model)

# ------

# Ajuste de promedios | Mínimos cuadrados | Post-Hoc

# 1. Separación de promedios
```

```r
library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Algoritmo)
pairs(marginal, adjust="tukey")

# 2. Visión compacta

library(multcomp)
CLD <- cld(marginal, alpha=0.05, Letters = letters, adjust = "tukey")
CLD

# 3. Gráfico promedios, intervalos de confianza y letras de separación

# Ordenamos los niveles para imprimirlos
CLD$Algoritmo <- factor(CLD$Algoritmo, levels = c("Algoritmo A", "Algoritmo B", "Algoritmo C"))
# Removemos espacios en blanco
CLD$.group <- gsub(" ", "", CLD$.group)

library(ggplot2)
ggplot(CLD,
       aes( x = Algoritmo,
            y = lsmean,
            label = .group)) +
       geom_point(shape = 15, size = 4) +
       geom_errorbar(aes(ymin = lower.CL,
                         ymax = upper.CL),
                         width = 0.2,
                         size = 0.7) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold"),
             axis.text = element_text(face = "bold"),
             plot.caption = element_text(hjust = 0)) +

       ylab("Promedio del minimo cuadrado \n
             Tiempo de ejecucion") +

       geom_text(nudge_x = c(0,0,0),
                 nudge_y = c(1100, 1100, 1100),
                 color = "black")

# Salvar gráficos

#svg("cajas1.svg")
#def.off()
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: Rmisc

Loading required package: lattice

Loading required package: plyr


Attaching package: 'plyr'


The following object is masked from 'package:FSA':

    mapvalues


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.
```

```
Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 3

	Algoritmo	Ejecucion	Tiempo
	<fct>	<chr>	<chr>
1	Algoritmo A	1	12060
2	Algoritmo A	2	14089
3	Algoritmo A	3	13502
4	Algoritmo A	4	9574
...	NA	...	...
57	Algoritmo C	17	9537
58	Algoritmo C	18	13014
59	Algoritmo C	19	14033
60	Algoritmo C	20	11093

```
'data.frame':   60 obs. of  3 variables:
 $ Algoritmo: Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Ejecucion: int  1 2 3 4 5 6 7 8 9 10 ...
 $ Tiempo   : int  12060 14089 13502 9574 14056 11569 13047 13275 14257 15075 ...
      Algoritmo    Ejecucion         Tiempo
 Algoritmo A:20   Min.   : 1.00   Min.   : 9081
 Algoritmo B:20   1st Qu.: 5.75   1st Qu.:11093
 Algoritmo C:20   Median :10.50   Median :12094
                  Mean   :10.50   Mean   :12234
                  3rd Qu.:15.25   3rd Qu.:13262
                  Max.   :20.00   Max.   :17098
```

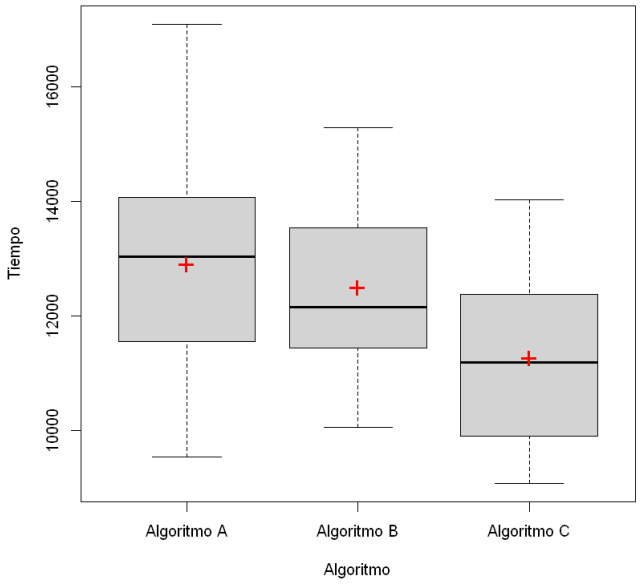
A data.frame: 3 × 9

Algoritmo	n	mean	sd	min	Q1	median	Q3	max
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	12918.80	1941.191	9548	11566.00	13043.5	14064.25	17098
Algoritmo B	20	12505.60	1414.667	10067	11510.50	12154.5	13544.00	15297
Algoritmo C	20	11276.15	1424.242	9081	10067.25	11193.5	12311.25	14033

A data.frame: 3 × 6

Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	12900	0.95	12100	13700
Algoritmo B	20	12500	0.95	11900	13100
Algoritmo C	20	11300	0.95	10700	11900

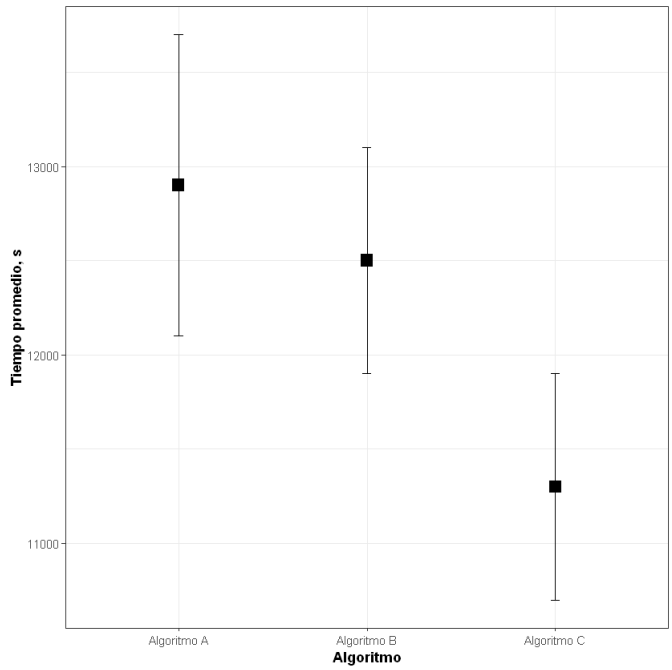
```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead."
```

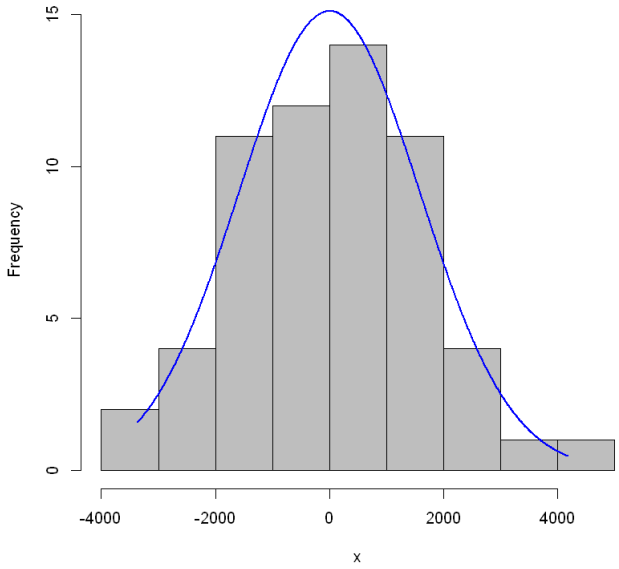


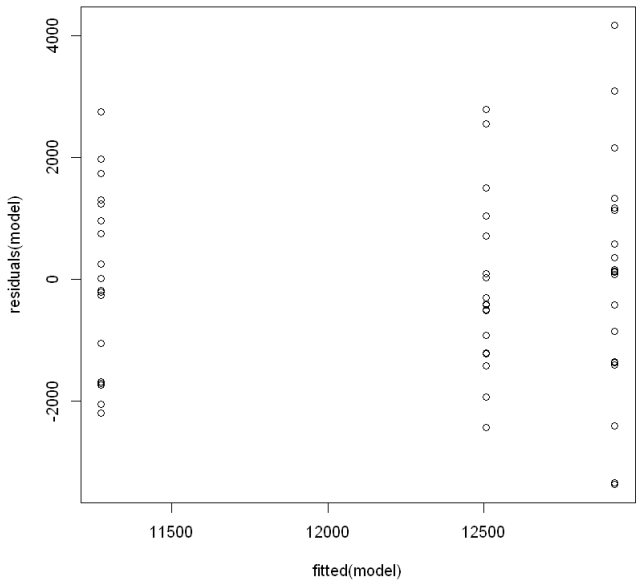

```
Call:
lm(formula = Tiempo ~ Algoritmo, data = Data)

Residuals:
    Min      1Q  Median      3Q     Max
-3370.8 -1211.6    25.1  1065.4  4179.2

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           12918.8      360.5  35.835  < 2e-16 ***
AlgoritmoAlgoritmo B   -413.2      509.8  -0.810  0.42105
AlgoritmoAlgoritmo C  -1642.7      509.8  -3.222  0.00211 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1612 on 57 degrees of freedom
Multiple R-squared:  0.1647,    Adjusted R-squared:  0.1353
F-statistic: 5.618 on 2 and 57 DF,  p-value: 0.005932
```

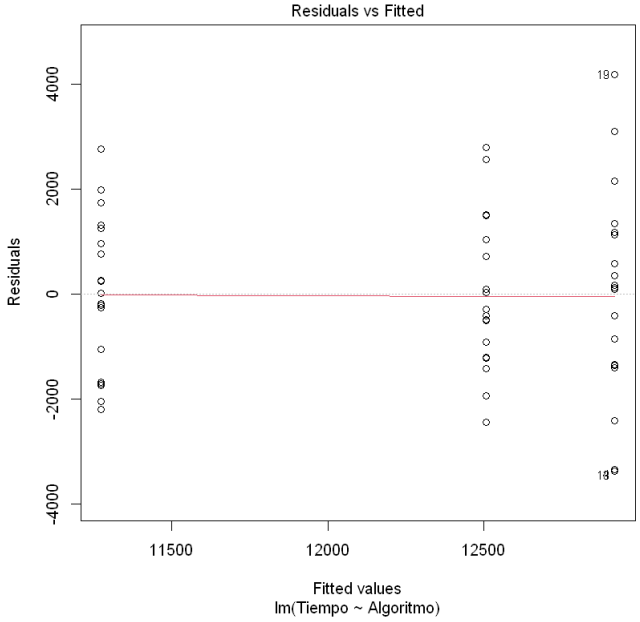
A anova: 2 × 4

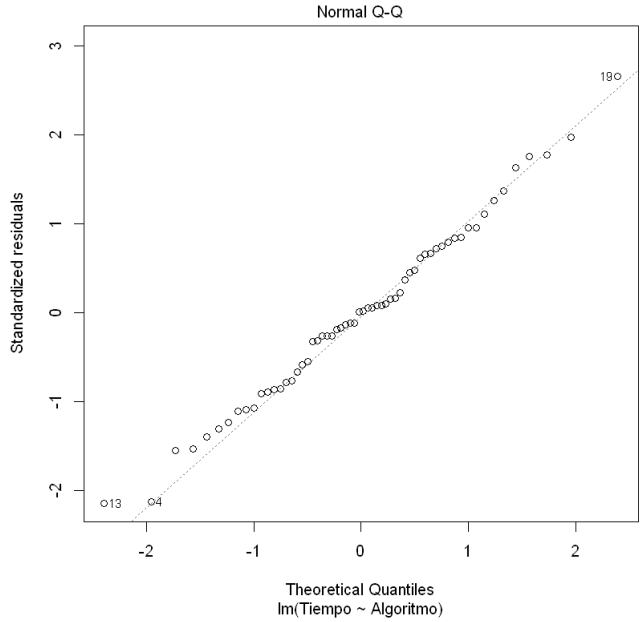
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo	29203870	2	5.617588	0.005931933
Residuals	148161499	57	NA	NA

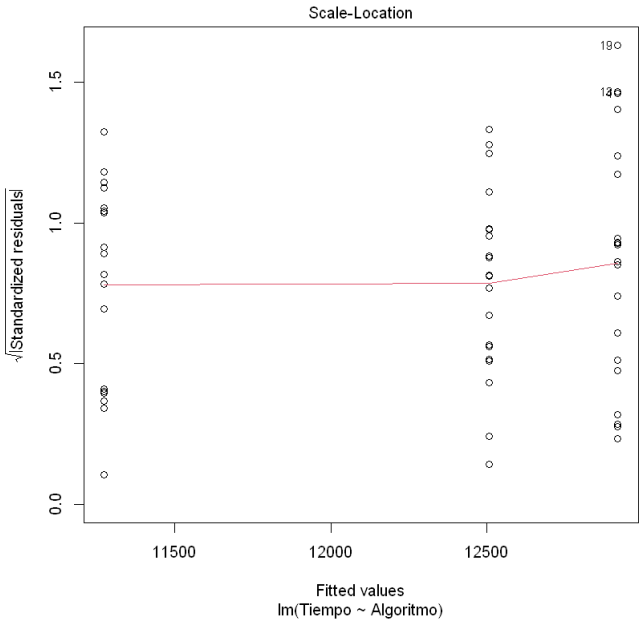



Frequency
0
5
10
15
-4000
-2000
0
2000
4000
x


residuals(model)
-2000
0
2000
4000
11500
12000
12500
fitted(model)


Residuals vs Fitted
Residuals
-4000
-2000
0
2000
4000
11500
12000
12500
Fitted values
lm(Tiempo ~ Algoritmo)

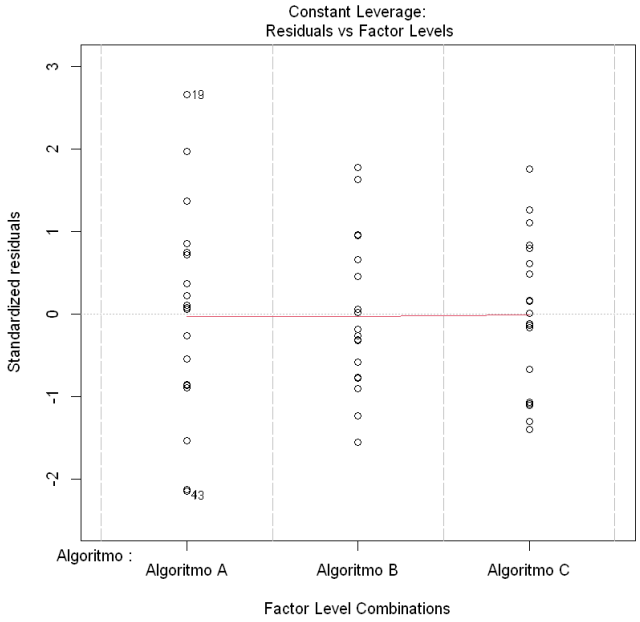


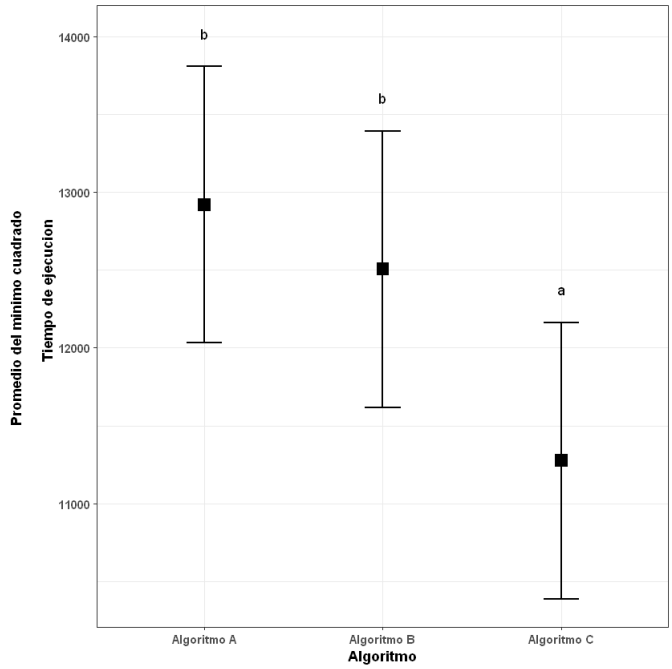



```
 contrast                 estimate  SE df t.ratio p.value
 Algoritmo A - Algoritmo B      413 510 57   0.810  0.6981
 Algoritmo A - Algoritmo C     1643 510 57   3.222  0.0059
 Algoritmo B - Algoritmo C     1229 510 57   2.411  0.0494

P value adjustment: tukey method for comparing a family of 3 estimates
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 3 × 7

	Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	Algoritmo C	11276.15	360.5083	57	10389.33	12162.97	a
2	Algoritmo B	12505.60	360.5083	57	11618.78	13392.42	b
1	Algoritmo A	12918.80	360.5083	57	12031.98	13805.62	b





In [ ]:

In [1]:

```
# ANOVA Monofactorial

# 1. Carga inicial de datos:

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(Rmisc)){install.packages("Rmisc")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(multcompView)){install.packages("multcomp")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


Datos <- ("
Algoritmo       Ejecucion  Tiempo
'Algoritmo A'      '1'    12060
'Algoritmo A'      '2'    14089
'Algoritmo A'      '3'    13502
'Algoritmo A'      '4'     9574
'Algoritmo A'      '5'    14056
'Algoritmo A'      '6'    11569
'Algoritmo A'      '7'    13047
'Algoritmo A'      '8'    13275
'Algoritmo A'      '9'    14257
'Algoritmo A'      '10'   15075
'Algoritmo A'      '11'   12506
'Algoritmo A'      '12'   11557
'Algoritmo A'      '13'    9548
'Algoritmo A'      '14'   11514
'Algoritmo A'      '15'   16015
'Algoritmo A'      '16'   13004
'Algoritmo A'      '17'   10510
'Algoritmo A'      '18'   13040
'Algoritmo A'      '19'   17098
'Algoritmo A'      '20'   13080
'Algoritmo B'      '1'    11080
'Algoritmo B'      '2'    12089
'Algoritmo B'      '3'    12538
'Algoritmo B'      '4'    10571
'Algoritmo B'      '5'    12010
'Algoritmo B'      '6'    12598
'Algoritmo B'      '7'    13543
'Algoritmo B'      '8'    13547
'Algoritmo B'      '9'    13217
'Algoritmo B'      '10'   15297
'Algoritmo B'      '11'   12210
'Algoritmo B'      '12'   11299
'Algoritmo B'      '13'   10067
'Algoritmo B'      '14'   11279
'Algoritmo B'      '15'   14006
'Algoritmo B'      '16'   12099
'Algoritmo B'      '17'   11581
'Algoritmo B'      '18'   14012
'Algoritmo B'      '19'   15069
'Algoritmo B'      '20'   12000
'Algoritmo C'      '1'     9081
'Algoritmo C'      '2'    11012
'Algoritmo C'      '3'    11529
'Algoritmo C'      '4'     9569
'Algoritmo C'      '5'    11092
'Algoritmo C'      '6'    11524
'Algoritmo C'      '7'    12522
'Algoritmo C'      '8'    12588
'Algoritmo C'      '9'    12241
'Algoritmo C'      '10'   13257
'Algoritmo C'      '11'   11294
'Algoritmo C'      '12'   10226
'Algoritmo C'      '13'    9591
'Algoritmo C'      '14'    9224
'Algoritmo C'      '15'   12033
'Algoritmo C'      '16'   11063
'Algoritmo C'      '17'    9537
'Algoritmo C'      '18'   13014
```

```
'Algoritmo C'      '19'   14033
'Algoritmo C'      '20'   11093
")

# Lectura de los datos
Data <- read.table(textConnection(Datos), header=TRUE)
# Ordenar los datos segun los ingresamos
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))

# 2. Lectura de datos / Verificación de lectura

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(Datos)

# 3. Resumen organizado

Summarize(Tiempo ~ Algoritmo, data = Data, digits = 4)

# 4. Diagrama de cajas

M <- tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

# 5. Información de promedios e intervalos de confianza

Sum <- groupwiseMean(Tiempo ~ Algoritmo, data = Data, conf = 0.95, digits = 3, traditional = FALSE, percenti
Sum


# 6. Gráficos de promedios e intervalos de confianza

library(ggplot2)
ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Tiempo promedio, s")

# 7. Modelo Lineal

model <- lm(Tiempo ~ Algoritmo, data = Data)
summary(model)

# EMPIEZA EL CAMBIO

# 8. Histograma de residuos

X <- residuals(model)
library(rcompanion)
plotNormalHistogram(X)

# 9. Dispersión de residuos

plot(fitted(model),residuals(model))

# 10. Gráficos del modelo lineal

plot(model)

# 11. ANOVA

library(car)
Anova(model, type = "II")


# ------
```

```r
# Ajuste de promedios | Mínimos cuadrados | Post-Hoc

# 1. Separación de promedios

library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Algoritmo)
pairs(marginal, adjust="tukey")

# 2. Visión compacta

library(multcomp)
CLD <- cld(marginal, alpha=0.05, Letters = letters, adjust = "tukey")
CLD

# 3. Gráfico promedios, intervalos de confianza y letras de separación

# Ordenamos los niveles para imprimirlos
CLD$Algoritmo <- factor(CLD$Algoritmo, levels = c("Algoritmo A", "Algoritmo B", "Algoritmo C"))
# Removemos espacios en blanco
CLD$.group <- gsub(" ", "", CLD$.group)

library(ggplot2)
ggplot(CLD,
       aes( x = Algoritmo,
            y = lsmean,
            label = .group)) +
       geom_point(shape = 15, size = 4) +
       geom_errorbar(aes(ymin = lower.CL,
                         ymax = upper.CL),
                         width = 0.2,
                         size = 0.7) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold"),
             axis.text = element_text(face = "bold"),
             plot.caption = element_text(hjust = 0)) +

       ylab("Promedio del minimo cuadrado \n
             Tiempo de ejecucion") +

       geom_text(nudge_x = c(0,0,0),
                 nudge_y = c(1100, 1100, 1100),
                 color = "black")
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: Rmisc

Loading required package: lattice

Loading required package: plyr


Attaching package: 'plyr'


The following object is masked from 'package:FSA':

    mapvalues


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.
```

```
Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 3

	Algoritmo	Ejecucion	Tiempo
	<fct>	<chr>	<chr>
1	Algoritmo A	1	12060
2	Algoritmo A	2	14089
3	Algoritmo A	3	13502
4	Algoritmo A	4	9574
...	NA	...	...
57	Algoritmo C	17	9537
58	Algoritmo C	18	13014
59	Algoritmo C	19	14033
60	Algoritmo C	20	11093

```
'data.frame':   60 obs. of  3 variables:
 $ Algoritmo: Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Ejecucion: int  1 2 3 4 5 6 7 8 9 10 ...
 $ Tiempo   : int  12060 14089 13502 9574 14056 11569 13047 13275 14257 15075 ...
      Algoritmo    Ejecucion         Tiempo
 Algoritmo A:20   Min.   : 1.00   Min.   : 9081
 Algoritmo B:20   1st Qu.: 5.75   1st Qu.:11093
 Algoritmo C:20   Median :10.50   Median :12094
                  Mean   :10.50   Mean   :12234
                  3rd Qu.:15.25   3rd Qu.:13262
                  Max.   :20.00   Max.   :17098
```

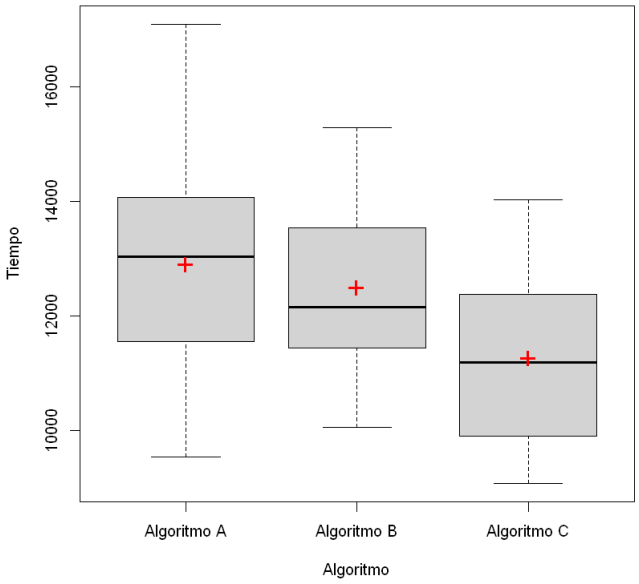
A data.frame: 3 × 9

Algoritmo	n	mean	sd	min	Q1	median	Q3	max
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	12918.80	1941.191	9548	11566.00	13043.5	14064.25	17098
Algoritmo B	20	12505.60	1414.667	10067	11510.50	12154.5	13544.00	15297
Algoritmo C	20	11276.15	1424.242	9081	10067.25	11193.5	12311.25	14033

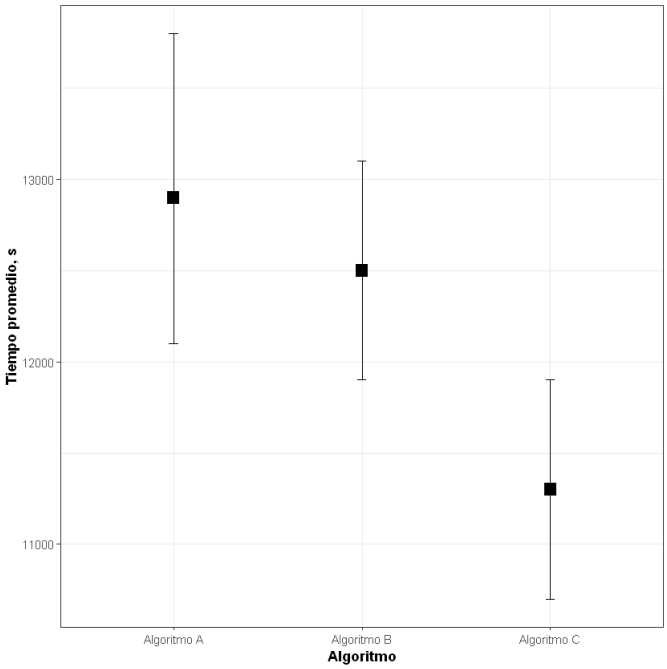
A data.frame: 3 × 6

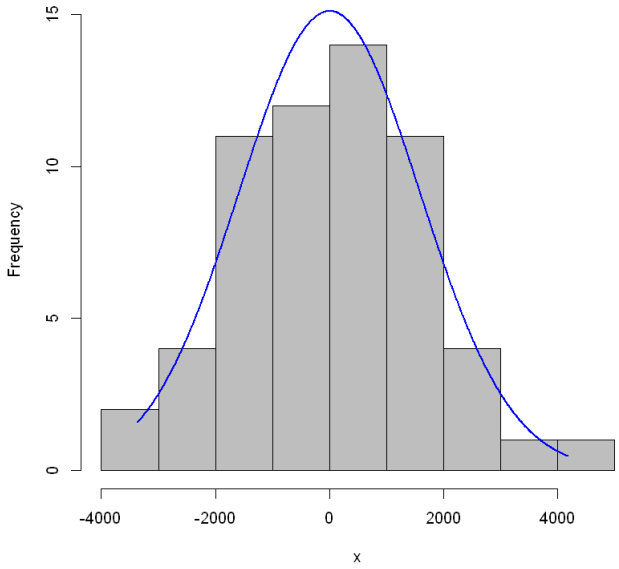
Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	12900	0.95	12100	13800
Algoritmo B	20	12500	0.95	11900	13100
Algoritmo C	20	11300	0.95	10700	11900

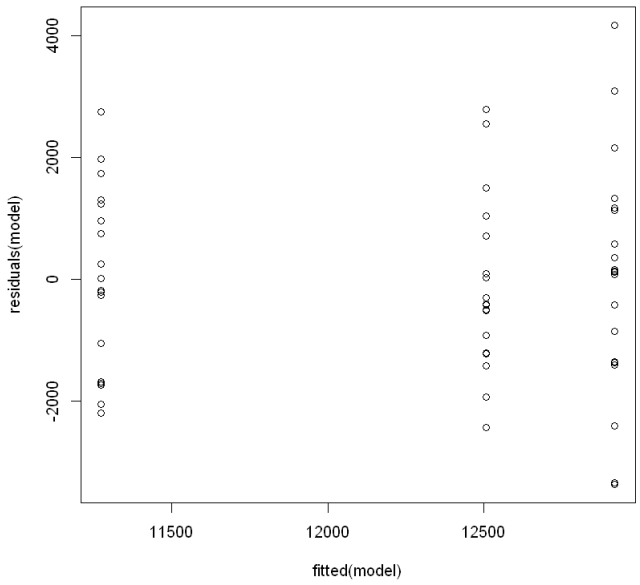
```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead."
```

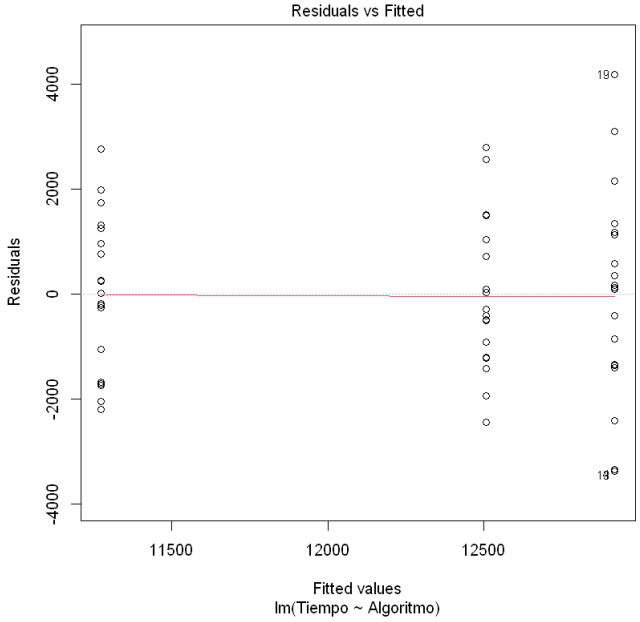


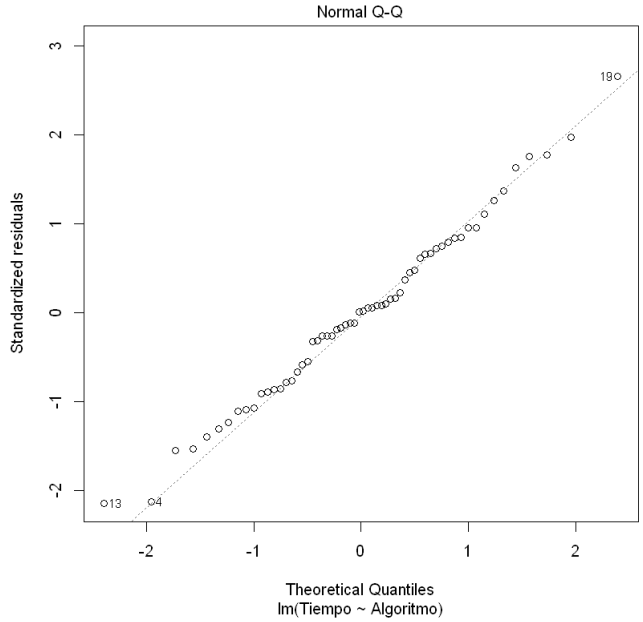

```
Call:
lm(formula = Tiempo ~ Algoritmo, data = Data)

Residuals:
    Min      1Q  Median      3Q     Max
-3370.8 -1211.6    25.1  1065.4  4179.2

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           12918.8      360.5  35.835  < 2e-16 ***
AlgoritmoAlgoritmo B   -413.2      509.8  -0.810  0.42105
AlgoritmoAlgoritmo C  -1642.7      509.8  -3.222  0.00211 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1612 on 57 degrees of freedom
Multiple R-squared:  0.1647,    Adjusted R-squared:  0.1353
F-statistic: 5.618 on 2 and 57 DF,  p-value: 0.005932
```

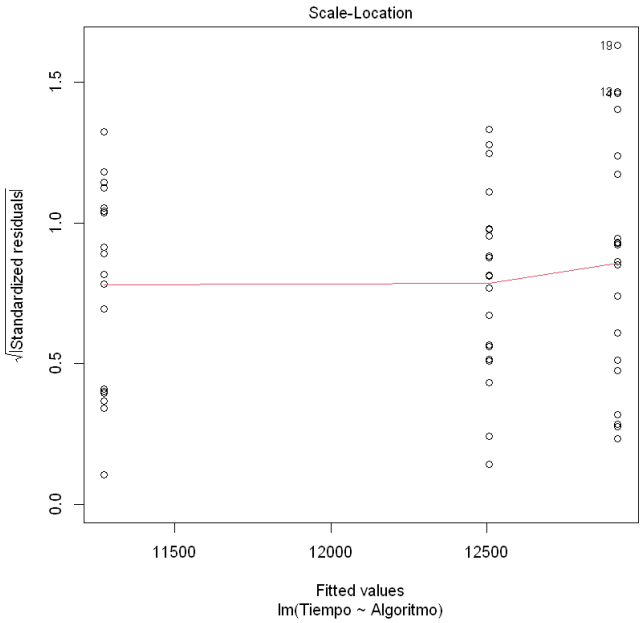



Frequency
0
5
10
15
-4000
-2000
0
2000
4000
x


residuals(model)
-2000
0
2000
4000
11500
12000
12500
fitted(model)


Residuals vs Fitted
Residuals
-4000
-2000
0
2000
4000
11500
12000
12500
Fitted values
lm(Tiempo ~ Algoritmo)





A anova: 2 × 4

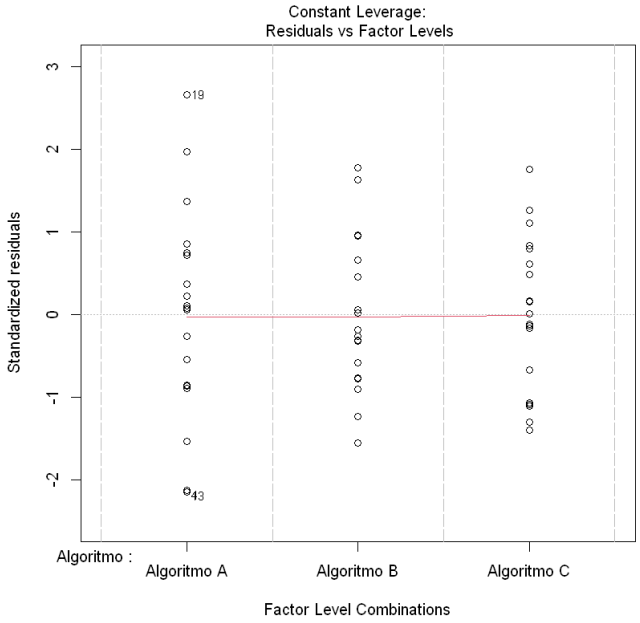
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo	29203870	2	5.617588	0.005931933
Residuals	148161499	57	NA	NA

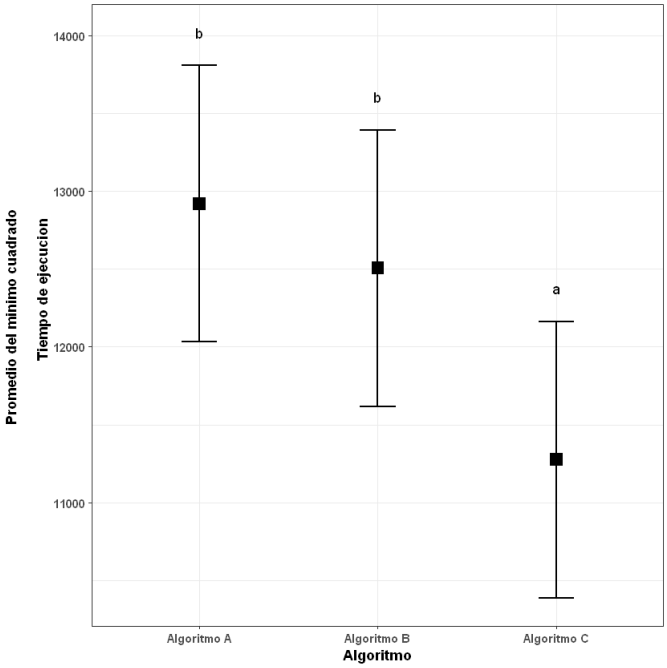
```
contrast                 estimate  SE df t.ratio p.value
Algoritmo A - Algoritmo B      413 510 57   0.810  0.6981
Algoritmo A - Algoritmo C     1643 510 57   3.222  0.0059
Algoritmo B - Algoritmo C     1229 510 57   2.411  0.0494

P value adjustment: tukey method for comparing a family of 3 estimates
```

```
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 3 × 7

	Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	Algoritmo C	11276.15	360.5083	57	10389.33	12162.97	a
2	Algoritmo B	12505.60	360.5083	57	11618.78	13392.42	b
1	Algoritmo A	12918.80	360.5083	57	12031.98	13805.62	b




Promedio del minimo cuadrado
Tiempo de ejecucion
14000
13000
12000
11000
b
b
a
Algoritmo A
Algoritmo B
Algoritmo C
Algoritmo

In [1]:

```
# Analisis Anova multifactorial

# 1. Carga inicial de datos.

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


ln <- ("Algoritmo       Entrenamiento       Rendimiento
'Algoritmo A'        MT500           12000
'Algoritmo A'        MT500           14005
'Algoritmo A'        MT500           13508
'Algoritmo A'        MT500            9503
'Algoritmo A'        MT500           14004
'Algoritmo A'        MT1000           11502
'Algoritmo A'        MT1000           13006
'Algoritmo A'        MT1000           13252
'Algoritmo A'        MT1000           14253
'Algoritmo A'        MT1000           15003
'Algoritmo A'        MT5000          12504
'Algoritmo A'        MT5000          11504
'Algoritmo A'        MT5000           9500
'Algoritmo A'        MT5000          11506
'Algoritmo A'        MT5000          16000
'Algoritmo A'        MT50000          13008
'Algoritmo A'        MT50000          10506
'Algoritmo A'        MT50000          13005
'Algoritmo A'        MT50000          17002
'Algoritmo A'        MT50000          13008
'Algoritmo B'        MT500            11005
'Algoritmo B'        MT500            12007
'Algoritmo B'        MT500            12509
'Algoritmo B'        MT500            10504
'Algoritmo B'        MT500            12002
'Algoritmo B'        MT1000           12504
'Algoritmo B'        MT1000           13501
'Algoritmo B'        MT1000           13501
'Algoritmo B'        MT1000           13252
'Algoritmo B'        MT1000           15256
'Algoritmo B'        MT5000           12253
'Algoritmo B'        MT5000           11255
'Algoritmo B'        MT5000           10006
'Algoritmo B'        MT5000           11252
'Algoritmo B'        MT5000           14004
'Algoritmo B'        MT50000          12007
'Algoritmo B'        MT50000          11505
'Algoritmo B'        MT50000          14009
'Algoritmo B'        MT50000          15000
'Algoritmo B'        MT50000          12009
'Algoritmo C'        MT500             9000
'Algoritmo C'        MT500            11003
'Algoritmo C'        MT500            11505
'Algoritmo C'        MT500             9509
'Algoritmo C'        MT500            11003
'Algoritmo C'        MT1000           11508
'Algoritmo C'        MT1000           12508
'Algoritmo C'        MT1000           12506
'Algoritmo C'        MT1000           12254
'Algoritmo C'        MT1000           13253
'Algoritmo C'        MT5000           11255
'Algoritmo C'        MT5000           10257
'Algoritmo C'        MT5000            9500
'Algoritmo C'        MT5000            9255
'Algoritmo C'        MT5000           12009
'Algoritmo C'        MT50000          11000
'Algoritmo C'        MT50000           9509
'Algoritmo C'        MT50000          13009
'Algoritmo C'        MT50000          14005
'Algoritmo C'        MT50000          11001
")
```

```r
# Se introduce la tabla.
Data <- read.table(textConnection(ln), header = TRUE)

# Se ordenan los datos según los ingresamos. (Evitar orden alfabético por R).
Data$Entrenamiento <- factor(Data$Entrenamiento, levels=unique(Data$Entrenamiento))

# 2. Verificación de la lectura de datos

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(ln)


# 3. Gráfico simple de interacción.

interaction.plot(x.factor = Data$Entrenamiento,
 trace.factor = Data$Algoritmo,
 response = Data$Rendimiento,
 fun = mean,
 type = "b",
 col = c("black", "red", "green"),
 pch = c(19,17,15),
 fixed = TRUE,
 leg.bty = "o")

# 4. Modelo lineal y ANOVA

model <- lm(Rendimiento ~ Entrenamiento + Algoritmo + Entrenamiento : Algoritmo, data = Data)

library(car)
Anova(model, type = "II")

# 5. Evaluación de supuestos

# Normalidad
x <- residuals(model)
library(rcompanion)
plotNormalHistogram(x)

# Disperción de los residuos
plot(fitted(model), residuals(model))

# Graficos del modelo lineal
plot(model)

# 6. Análisis post-hoc

library(lsmeans)
marginal <- lsmeans(model, pairwise ~ Algoritmo, adjust = "tukey")
marginal

# Funcion cld
library(multcomp)
CLD <- cld(marginal, alpha=0.05, Letters= letters, adjust="tukey")
CLD

# Análisis post-hoc entrenamiento
marginal <- lsmeans(model, pairwise ~ Entrenamiento, adjust = "tukey")
marginal

# Funcion cld
library(multcomp)
CLD <- cld(marginal, alpha=0.05, Letters= letters, adjust="tukey")
CLD


# 7. Gráfico final
library(FSA)

Sum <- Summarize(Rendimiento ~ Entrenamiento + Algoritmo, data = Data, digits = 3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum
```

```r
Sum$Entrenamiento <- factor(Sum$Entrenamiento,
 levels = unique(Sum$Entrenamiento))

# 8. Boxplot error estándar

library(FSA)
library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum, aes(x=Entrenamiento,
 y = mean,
 color = Algoritmo)) +
 geom_errorbar(aes(ymin=mean-se,
 ymax=mean + se),
 width=.2, size=0.7, position=pd) +
 geom_point(shape=15, size=4, position = pd) +
 theme_bw() +
 theme(axis.title = element_text(face="bold")) +
 scale_colour_manual(values = c("black", "red", "green")) +
 ylab("Rendimiento")
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

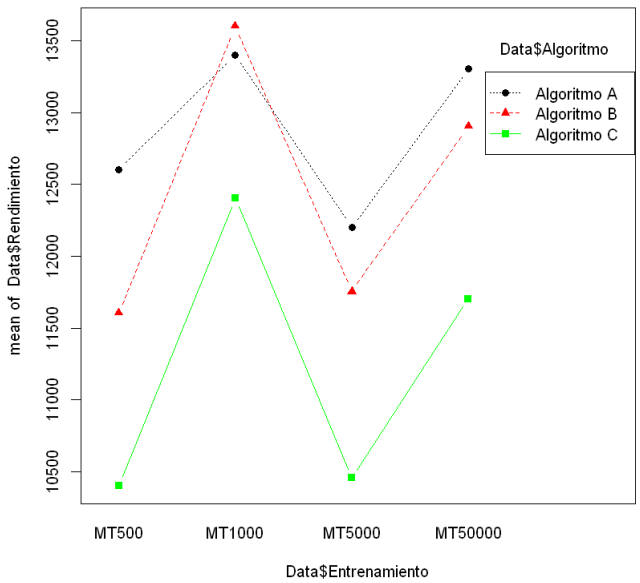
A data.frame: 9 × 3

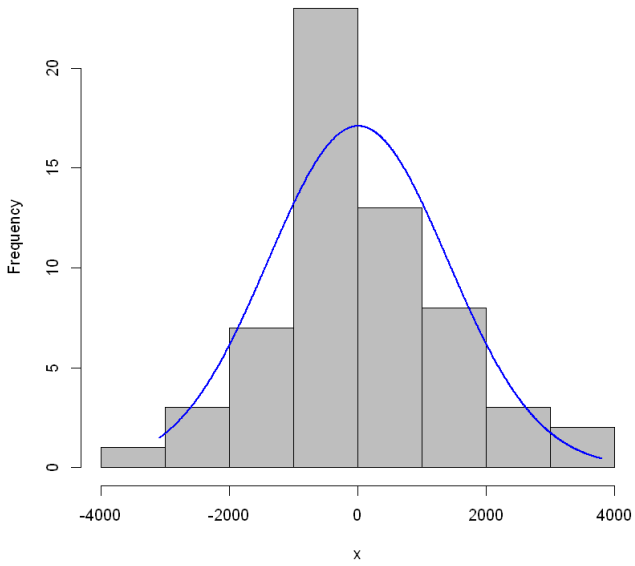
	Algoritmo	Entrenamiento	Rendimiento
	<chr>	<fct>	<chr>
1	Algoritmo A	MT500	12000
2	Algoritmo A	MT500	14005
3	Algoritmo A	MT500	13508
4	Algoritmo A	MT500	9503
...	NA	NA	...
57	Algoritmo C	MT50000	9509
58	Algoritmo C	MT50000	13009
59	Algoritmo C	MT50000	14005
60	Algoritmo C	MT50000	11001

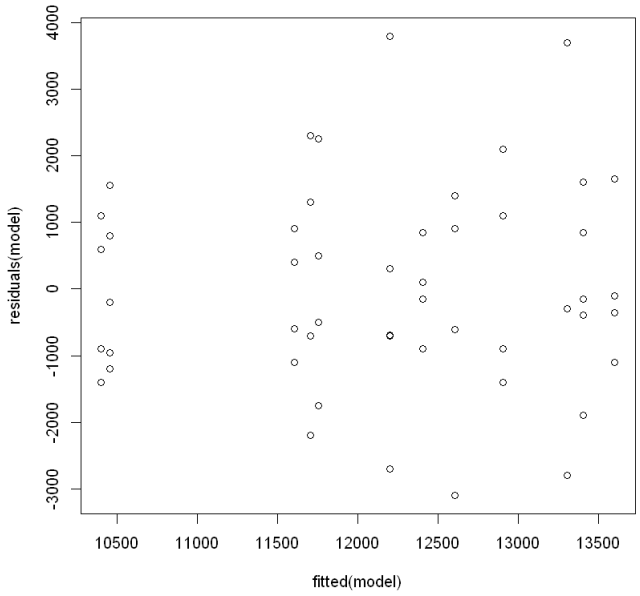
```
'data.frame':   60 obs. of  3 variables:
 $ Algoritmo    : chr  "Algoritmo A" "Algoritmo A" "Algoritmo A" "Algoritmo A" ...
 $ Entrenamiento: Factor w/ 4 levels "MT500","MT1000",..: 1 1 1 1 1 2 2 2 2 2 ...
 $ Rendimiento  : int  12000 14005 13508 9503 14004 11502 13006 13252 14253 15003 ...
  Algoritmo         Entrenamiento  Rendimiento
 Length:60          MT500  :15    Min.   : 9000
 Class :character   MT1000 :15    1st Qu.:11004
 Mode  :character   MT5000 :15    Median :12009
                    MT50000:15    Mean   :12196
                                  3rd Qu.:13252
                                  Max.   :17002
```

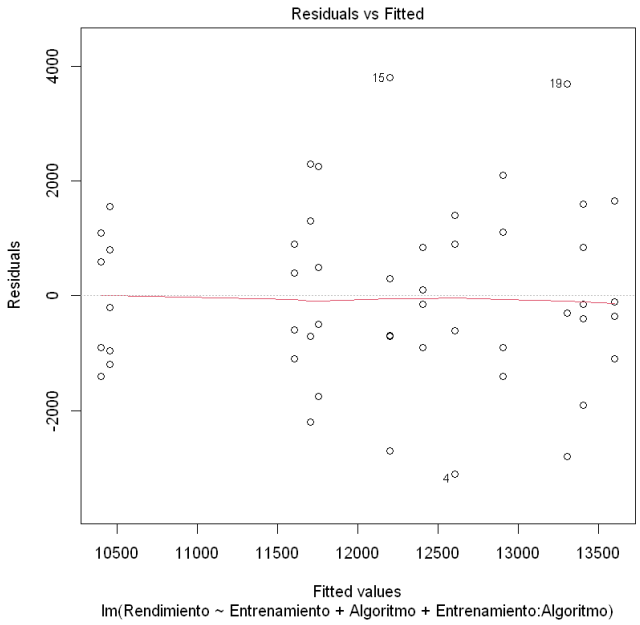
A anova: 4 × 4

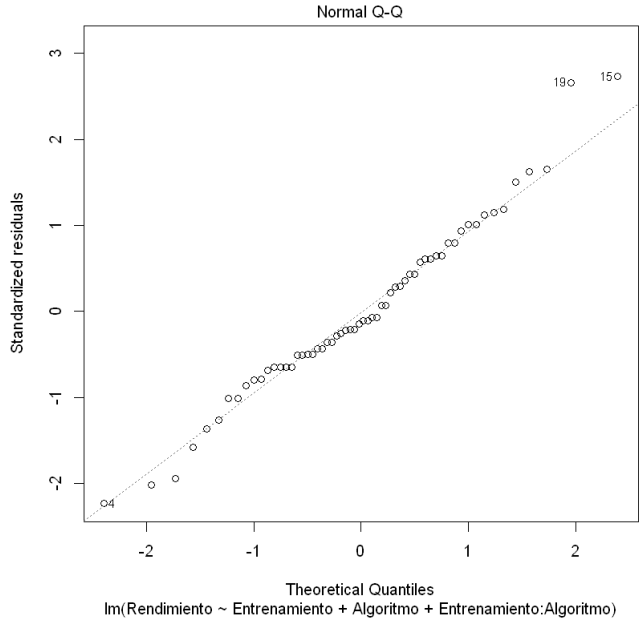
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Entrenamiento	30621741	3	4.2341389	0.009822338
Algoritmo	28982927	2	6.0113044	0.004679860
Entrenamiento:Algoritmo	2444325	6	0.1689912	0.983866401
Residuals	115713695	48	NA	NA

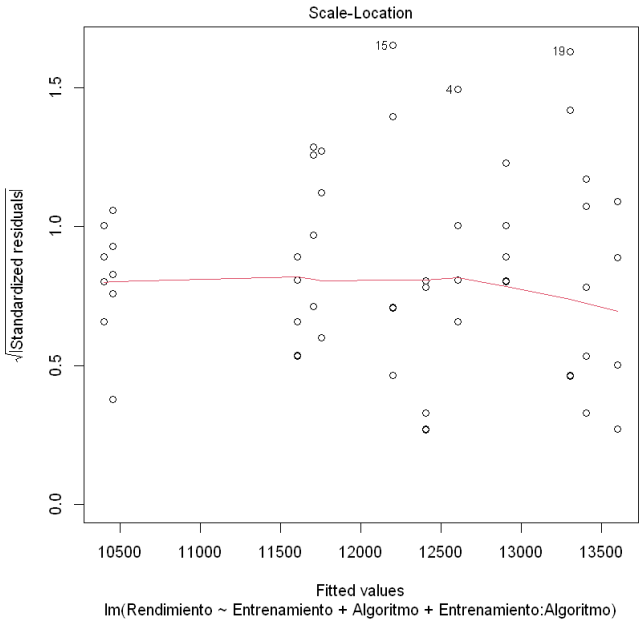



Frequency
0
5
10
15
20
-4000
-2000
0
2000
4000
x


residuals(model)
-3000
-2000
-1000
0
1000
2000
3000
4000
10500
11000
11500
12000
12500
13000
13500
fitted(model)


Residuals vs Fitted
Residuals
-2000
0
2000
4000
15
19
4
10500
11000
11500
12000
12500
13000
13500
Fitted values
lm(Rendimiento ~ Entrenamiento + Algoritmo + Entrenamiento:Algoritmo)






```
NOTE: Results may be misleading due to involvement in interactions
```

$lsmeans

$bhat

$V

$levels

$linfct

$dffun

$dfargs
$post.beta

$estName
$estType
$infer

$level
$adjust
$famSize
$avgd.over
$sigma
$methDesc
$extras

$contrasts

$bhat

.wgt.
<dbl>
20
20
20

$V

	(Intercept)	En
(Intercept)	482140.4	
EntrenamientoMT1000	-482140.4	
EntrenamientoMT5000	-482140.4	
EntrenamientoMT50000	-482140.4	
AlgoritmoAlgoritmo B	-482140.4	
AlgoritmoAlgoritmo C	-482140.4	
EntrenamientoMT1000:AlgoritmoAlgoritmo B	482140.4	
EntrenamientoMT5000:AlgoritmoAlgoritmo B	482140.4	
EntrenamientoMT50000:AlgoritmoAlgoritmo B	482140.4	
EntrenamientoMT1000:AlgoritmoAlgoritmo C	482140.4	
EntrenamientoMT5000:AlgoritmoAlgoritmo C	482140.4	
EntrenamientoMT50000:AlgoritmoAlgoritmo C	482140.4	

$levels

$contrast =
'Algoritmo A - Algoritmo B' · 'Algoritmo A - Algoritmo C' ·

$linfct

(Intercept)	EntrenamientoMT1000	EntrenamientoMT5000
0	0	0
0	0	0
0	0	0

$dffun

```
function (k, dfargs)
dfargs$df
```

$dfargs
$df = 48

$post.beta
A
matrix:
1 × 1
of
type
lgl
NA

$estName
'estimate'

$estType
'pairs'

$infer
FALSE · TRUE

$level
0.95

$adjust
'tukey'

$famSize
3

$avgd.over
'Entrenamiento'

$methDesc
'pairwise differences'

$is.new.rg FALSE

$.pairby ''

$orig.grid

A data.frame: 3 × 1

Algoritmo <fct>
Algoritmo A
Algoritmo B
Algoritmo C

$con.coef

A matrix: 3 × 3 of type dbl

	Algoritmo A	Algoritmo B	Algo
Algoritmo A - Algoritmo B	1	-1	
Algoritmo A - Algoritmo C	1	0	
Algoritmo B - Algoritmo C	0	1	

```
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 3 × 7

	Algoritmo <fct>	lsmean <dbl>	SE <dbl>	df <dbl>	lower.CL <dbl>	upper.CL <dbl>	.group <chr>
3	Algoritmo C	11242.45	347.1817	48	10383.55	12101.35	a
2	Algoritmo B	12467.05	347.1817	48	11608.15	13325.95	b
1	Algoritmo A	12878.95	347.1817	48	12020.05	13737.85	b

```
NOTE: Results may be misleading due to involvement in interactions
```

$lsmeans

$bhat

$V

$levels

$linfct

$dffun

$dfargs
$post.beta

$estName
$estType
$infer

$level
$adjust
$famSize
$avgd.over
$sigma
$methDesc
$extras

.wgt.
<dbl>
15
15
15
15

$bhat

12604 · 799.199999999997 · -401.200000000003 · 701.799

$V

	(Intercept)	En
(Intercept)	482140.4	
EntrenamientoMT1000	-482140.4	
EntrenamientoMT5000	-482140.4	
EntrenamientoMT50000	-482140.4	
AlgoritmoAlgoritmo B	-482140.4	
AlgoritmoAlgoritmo C	-482140.4	
EntrenamientoMT1000:AlgoritmoAlgoritmo B	482140.4	
EntrenamientoMT5000:AlgoritmoAlgoritmo B	482140.4	
EntrenamientoMT50000:AlgoritmoAlgoritmo B	482140.4	
EntrenamientoMT1000:AlgoritmoAlgoritmo C	482140.4	
EntrenamientoMT5000:AlgoritmoAlgoritmo C	482140.4	
EntrenamientoMT50000:AlgoritmoAlgoritmo C	482140.4	

$levels

$contrast =
'MT500 - MT1000' · 'MT500 - MT5000' · 'MT500 - MT50000

$linfct

(Intercept)	EntrenamientoMT1000	EntrenamientoMT5000
0	-1	0
0	0	-1
0	0	0
0	1	-1
0	1	0
0	0	1

$dffun

```
function (k, dfargs)
dfargs$df
```

$dfargs

$df = 48

$post.beta

A matrix: 1 × 1 of type lgl

NA

$estName

'estimate'

$estType

'pairs'

$infer FALSE · TRUE

$level 0.95
$adjust 'tukey'
$famSize 4
$avgd.over 'Algoritmo'
$methDesc 'pairwise differences'
$is.new.rg FALSE
$.pairby ''
$orig.grid A data.frame: 4 × 1

Entrenamiento
<fct>
MT500
MT1000
MT5000
MT50000

$con.coef A matrix: 6 × 4 of type dbl

	MT500	MT1000	MT5000	MT50000
MT500 - MT1000	1	-1	0	0
MT500 - MT5000	1	0	-1	0
MT500 - MT50000	1	0	0	-1
MT1000 - MT5000	0	1	-1	0
MT1000 - MT50000	0	1	0	-1
MT5000 - MT50000	0	0	1	-1

```
Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

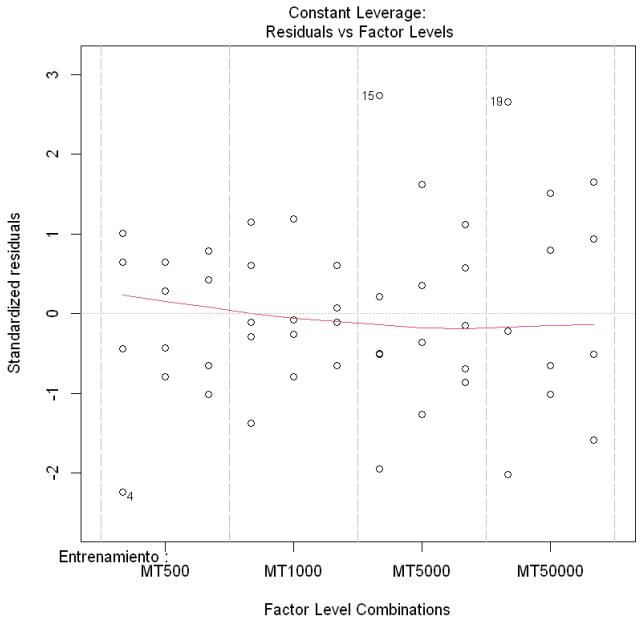
A summary_emm: 4 × 7

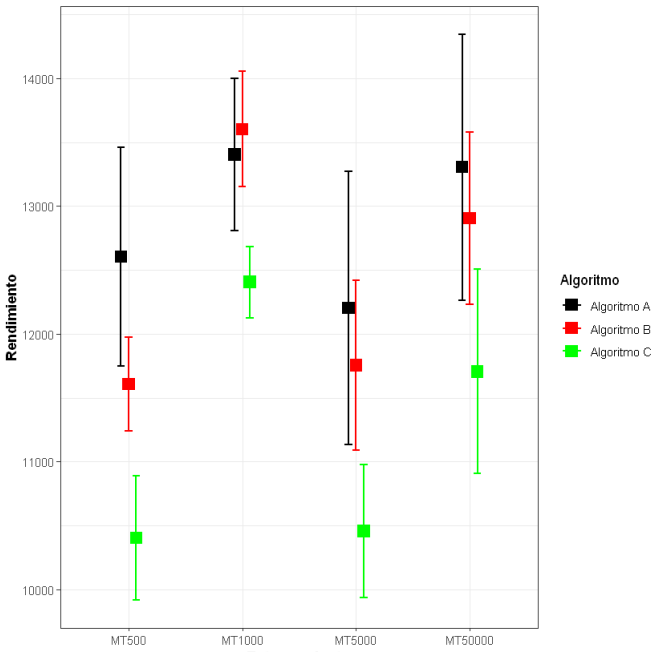
	Entrenamiento	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	MT5000	11470.67	400.8908	48	10433.24	12508.09	a
1	MT500	11537.80	400.8908	48	10500.38	12575.22	a
4	MT50000	12638.87	400.8908	48	11601.44	13676.29	ab
2	MT1000	13137.27	400.8908	48	12099.84	14174.69	b

A data.frame: 12 × 11

Entrenamiento	Algoritmo	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
MT500	Algoritmo A	5	12604.0	1918.400	9503	12000	13508	14004	14005	858
MT1000	Algoritmo A	5	13403.2	1330.104	11502	13006	13252	14253	15003	595
MT5000	Algoritmo A	5	12202.8	2386.861	9500	11504	11506	12504	16000	1070
MT50000	Algoritmo A	5	13305.8	2332.843	10506	13005	13008	13008	17002	1040
MT500	Algoritmo B	5	11605.4	822.896	10504	11005	12002	12007	12509	368
MT1000	Algoritmo B	5	13602.8	1010.473	12504	13252	13501	13501	15256	452
MT5000	Algoritmo B	5	11754.0	1488.960	10006	11252	11255	12253	14004	666
MT50000	Algoritmo B	5	12906.0	1514.683	11505	12007	12009	14009	15000	677
MT500	Algoritmo C	5	10404.0	1084.210	9000	9509	11003	11003	11505	485
MT1000	Algoritmo C	5	12405.8	625.963	11508	12254	12506	12508	13253	280
MT5000	Algoritmo C	5	10455.2	1166.849	9255	9500	10257	11255	12009	522
MT50000	Algoritmo C	5	11704.8	1789.237	9509	11000	11001	13009	14005	800

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```




Rendimiento
14000
13000
12000
11000
10000
MT500
MT1000
MT5000
MT50000
Algoritmo
Algoritmo A
Algoritmo B
Algoritmo C

In [ ]:

In [1]:

```
# ANOVA Monofactorial

# 1. Carga inicial de datos:

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(Rmisc)){install.packages("Rmisc")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(multcompView)){install.packages("multcomp")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}

Datos <- ("
Algoritmo      Ejecucion    Puntaje
'ERA'            '1'         45033
'ERA'            '2'         46623
'ERA'            '3'         43845
'ERA'            '4'         48849
'ERA'            '5'         45471
'ERA'            '6'         47132
'ERA'            '7'         46175
'ERA'            '8'         44015
'ERA'            '9'         46189
'ERA'            '10'        48499
'ERA'            '11'        42445
'ERA'            '12'        49155
'ERA'            '13'        48019
'ERA'            '14'        49068
'ERA'            '15'        42040
'ERA'            '16'        42538
'ERA'            '17'        44734
'ERA'            '18'        49899
'ERA'            '19'        47471
'ERA'            '20'        42966
'ERA'            '21'        42895
'ERA'            '22'        49284
'ERA'            '23'        45463
'ERA'            '24'        48812
'ERA'            '25'        43817
'ERA'            '26'        42326
'ERA'            '27'        43323
'ERA'            '28'        43482
'ERA'            '29'        44474
'ERA'            '30'        48576
'ERA'            '31'        42984
'ERA'            '32'        42914
'ERA'            '33'        48492
'ERA'            '34'        44776
'ERA'            '35'        48997
'ERA'            '36'        42966
'ERA'            '37'        42632
'ERA'            '38'        48334
'ERA'            '39'        45233
'ERA'            '40'        43456
'ERA'            '41'        46802
'ERA'            '42'        45332
'ERA'            '43'        43422
'ERA'            '44'        46946
'ERA'            '45'        42401
'ERA'            '46'        43473
'ERA'            '47'        45527
'ERA'            '48'        42785
'ERA'            '49'        47040
'ERA'            '50'        46662
'ERA'            '51'        49270
'ERA'            '52'        45591
'ERA'            '53'        46501
'ERA'            '54'        48277
'ERA'            '55'        47178
'ERA'            '56'        47658
'ERA'            '57'        49259
'ERA'            '58'        46043
'ERA'            '59'        46578
```

```
'ERA'           '60'        45165
'CFS'           '1'         42906
'CFS'           '2'         43517
'CFS'           '3'         42032
'CFS'           '4'         41393
'CFS'           '5'         42820
'CFS'           '6'         44951
'CFS'           '7'         41741
'CFS'           '8'         45950
'CFS'           '9'         42535
'CFS'           '10'        44958
'CFS'           '11'        44690
'CFS'           '12'        40945
'CFS'           '13'        44157
'CFS'           '14'        44550
'CFS'           '15'        42781
'CFS'           '16'        43145
'CFS'           '17'        43578
'CFS'           '18'        44312
'CFS'           '19'        45834
'CFS'           '20'        44558
'CFS'           '21'        42529
'CFS'           '22'        44373
'CFS'           '23'        46034
'CFS'           '24'        42572
'CFS'           '25'        41411
'CFS'           '26'        45356
'CFS'           '27'        44186
'CFS'           '28'        43339
'CFS'           '29'        45815
'CFS'           '30'        43666
'CFS'           '31'        45324
'CFS'           '32'        45427
'CFS'           '33'        41425
'CFS'           '34'        43171
'CFS'           '35'        40805
'CFS'           '36'        41931
'CFS'           '37'        40793
'CFS'           '38'        41542
'CFS'           '39'        45018
'CFS'           '40'        41054
'CFS'           '41'        44277
'CFS'           '42'        45672
'CFS'           '43'        46409
'CFS'           '44'        43394
'CFS'           '45'        43966
'CFS'           '46'        46300
'CFS'           '47'        46196
'CFS'           '48'        42279
'CFS'           '49'        45212
'CFS'           '50'        43758
'CFS'           '51'        43222
'CFS'           '52'        41552
'CFS'           '53'        41354
'CFS'           '54'        45633
'CFS'           '55'        44274
'CFS'           '56'        41765
'CFS'           '57'        45777
'CFS'           '58'        45205
'CFS'           '59'        42317
'CFS'           '60'        41959
'ULE'           '1'         45512
'ULE'           '2'         42095
'ULE'           '3'         41297
'ULE'           '4'         43138
'ULE'           '5'         40823
'ULE'           '6'         43642
'ULE'           '7'         40638
'ULE'           '8'         44984
'ULE'           '9'         43633
'ULE'           '10'        42653
'ULE'           '11'        41374
'ULE'           '12'        41558
'ULE'           '13'        41849
'ULE'           '14'        45989
'ULE'           '15'        42002
```

```
'ULE'           '16'        44388
'ULE'           '17'        41622
'ULE'           '18'        42974
'ULE'           '19'        44685
'ULE'           '20'        44343
'ULE'           '21'        43707
'ULE'           '22'        45049
'ULE'           '23'        42358
'ULE'           '24'        40590
'ULE'           '25'        45059
'ULE'           '26'        40714
'ULE'           '27'        42034
'ULE'           '28'        45881
'ULE'           '29'        40350
'ULE'           '30'        41908
'ULE'           '31'        41146
'ULE'           '32'        40614
'ULE'           '33'        40719
'ULE'           '34'        40340
'ULE'           '35'        42265
'ULE'           '36'        45164
'ULE'           '37'        44624
'ULE'           '38'        45495
'ULE'           '39'        44174
'ULE'           '40'        40461
'ULE'           '41'        44851
'ULE'           '42'        45542
'ULE'           '43'        42928
'ULE'           '44'        45616
'ULE'           '45'        45940
'ULE'           '46'        45587
'ULE'           '47'        45573
'ULE'           '48'        41625
'ULE'           '49'        40204
'ULE'           '50'        43319
'ULE'           '51'        41055
'ULE'           '52'        40471
'ULE'           '53'        43741
'ULE'           '54'        43889
'ULE'           '55'        40335
'ULE'           '56'        40163
'ULE'           '57'        45193
'ULE'           '58'        40631
'ULE'           '59'        40973
'ULE'           '60'        42638
'Monotonic'     '1'         32490
'Monotonic'     '2'         39163
'Monotonic'     '3'         32927
'Monotonic'     '4'         38382
'Monotonic'     '5'         33590
'Monotonic'     '6'         39663
'Monotonic'     '7'         32283
'Monotonic'     '8'         32621
'Monotonic'     '9'         37613
'Monotonic'     '10'        37005
'Monotonic'     '11'        35527
'Monotonic'     '12'        39317
'Monotonic'     '13'        32426
'Monotonic'     '14'        36812
'Monotonic'     '15'        32478
'Monotonic'     '16'        35713
'Monotonic'     '17'        37565
'Monotonic'     '18'        32738
'Monotonic'     '19'        38524
'Monotonic'     '20'        33706
'Monotonic'     '21'        39618
'Monotonic'     '22'        34218
'Monotonic'     '23'        35823
'Monotonic'     '24'        35597
'Monotonic'     '25'        39642
'Monotonic'     '26'        33650
'Monotonic'     '27'        33173
'Monotonic'     '28'        33812
'Monotonic'     '29'        38799
'Monotonic'     '30'        36139
'Monotonic'     '31'        32847
```

```
'Monotonic'      '32'        39100
'Monotonic'      '33'        35042
'Monotonic'      '34'        38256
'Monotonic'      '35'        39075
'Monotonic'      '36'        36629
'Monotonic'      '37'        35159
'Monotonic'      '38'        38597
'Monotonic'      '39'        34461
'Monotonic'      '40'        35573
'Monotonic'      '41'        38843
'Monotonic'      '42'        34925
'Monotonic'      '43'        33918
'Monotonic'      '44'        33043
'Monotonic'      '45'        36867
'Monotonic'      '46'        33323
'Monotonic'      '47'        38749
'Monotonic'      '48'        39796
'Monotonic'      '49'        37803
'Monotonic'      '50'        38739
'Monotonic'      '51'        36559
'Monotonic'      '52'        38500
'Monotonic'      '53'        35794
'Monotonic'      '54'        39715
'Monotonic'      '55'        38674
'Monotonic'      '56'        35441
'Monotonic'      '57'        34091
'Monotonic'      '58'        32393
'Monotonic'      '59'        32975
'Monotonic'      '60'        38212
")

# Lectura de los datos
Data <- read.table(textConnection(Datos), header=TRUE)
# Ordenar los datos segun los ingresamos
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))

# 2. Lectura de datos / Verificación de lectura

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(Datos)

# 3. Resumen organizado

Summarize(Puntaje ~ Algoritmo, data = Data, digits = 4)

# 4. Diagrama de cajas

M <- tapply(Data$Puntaje, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Puntaje ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

# 5. Información de promedios e intervalos de confianza

Sum <- groupwiseMean(Puntaje ~ Algoritmo, data = Data, conf = 0.95, digits = 3, traditional = FALSE, percent
Sum

# 6. Gráficos de promedios e intervalos de confianza

library(ggplot2)
ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Puntaje promedio, s")

#  Validacion de supuestos de ANOVA
#  Supuesto de normalidad y homocedasticidad
```

```r
# 7. Modelo Lineal

model <- lm(Puntaje ~ Algoritmo, data = Data)
summary(model)

# 8. Histograma de residuos

X <- residuals(model)
library(rcompanion)
dev.new()
#  Para evitar error "figure margins too large"
windows.options(width = 10, height = 8)
plotNormalHistogram(X)
#  Los residuos son normales

# 9. Dispersión de residuos

plot(fitted(model),residuals(model))
#  La dispersion es la misma

# 10. Gráficos del modelo lineal

plot(model)

# Se cumplen los supuestos

# 11. ANOVA

library(car)
Anova(model, type = "II")

# P-Value < alpha -> Se rechaza H0

# ------

# Ajuste de promedios | Mínimos cuadrados | Post-Hoc

# 1. Separación de promedios

library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Algoritmo)
pairs(marginal, adjust="tukey")

# 2. Visión compacta

library(multcomp)
CLD <- cld(marginal, alpha=0.05, Letters = letters, adjust = "tukey")
CLD

# Ordenamos los niveles para imprimirlos
CLD$Algoritmo <- factor(CLD$Algoritmo, levels = c("ERA", "CFS", "ULE", "Monotonic"))
# Removemos espacios en blanco
CLD$.group <- gsub(" ", "", CLD$.group)

# Era estadisticamente distinto a CFS, ULE y Monotonic
# CFS y ULE estadisticamente equivalentes.
# Monotonic estadisticamente distinto a CFS, ULE y Monotonic

library(ggplot2)
ggplot(CLD,
       aes( x = Algoritmo,
            y = lsmean,
            label = .group)) +
       geom_point(shape = 15, size = 4) +
       geom_errorbar(aes(ymin = lower.CL,
                         ymax = upper.CL),
                         width = 0.2,
                         size = 0.7) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold"),
             axis.text = element_text(face = "bold"),
             plot.caption = element_text(hjust = 0)) +

       ylab("Promedio del minimo cuadrado \n
             Tiempo de ejecucion") +
```

```
      geom_text(nudge_x = c(0,0,0),
                nudge_y = c(1100, 1100, 1100),
                color = "black")
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: Rmisc

Loading required package: lattice

Loading required package: plyr


Attaching package: 'plyr'


The following object is masked from 'package:FSA':

    mapvalues


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.
```

```
Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 3

	Algoritmo	Ejecucion	Puntaje
	<fct>	<chr>	<chr>
1	ERA	1	45033
2	ERA	2	46623
3	ERA	3	43845
4	ERA	4	48849
...	NA	...	...
237	Monotonic	57	34091
238	Monotonic	58	32393
239	Monotonic	59	32975
240	Monotonic	60	38212

```
'data.frame':   240 obs. of  3 variables:
 $ Algoritmo: Factor w/ 4 levels "ERA","CFS","ULE",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Ejecucion: int  1 2 3 4 5 6 7 8 9 10 ...
 $ Puntaje  : int  45033 46623 43845 48849 45471 47132 46175 44015 46189 48499 ...
    Algoritmo    Ejecucion        Puntaje
 ERA       :60   Min.   : 1.00   Min.   :32283
 CFS       :60   1st Qu.:15.75   1st Qu.:40071
 ULE       :60   Median :30.50   Median :42858
 Monotonic:60   Mean   :30.50   Mean   :42080
                3rd Qu.:45.25   3rd Qu.:45172
                Max.   :60.00   Max.   :49899
```

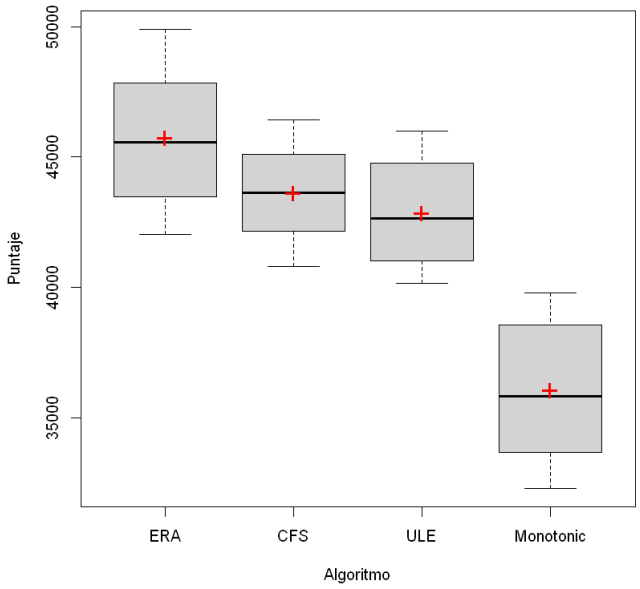
A data.frame: 4 × 9

Algoritmo	n	mean	sd	min	Q1	median	Q3	max
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
ERA	60	45755.20	2343.695	42040	43468.75	45559.0	47748.25	49899
CFS	60	43627.42	1664.518	40793	42217.25	43622.0	45064.75	46409
ULE	60	42868.75	1929.409	40163	41034.50	42645.5	44726.50	45989
Monotonic	60	36068.55	2503.939	32283	33692.00	35808.5	38542.25	39796

A data.frame: 4 × 6

Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
ERA	60	45800	0.95	45200	46300
CFS	60	43600	0.95	43200	44000
ULE	60	42900	0.95	42400	43400
Monotonic	60	36100	0.95	35400	36700

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```




```
Call:
lm(formula = Puntaje ~ Algoritmo, data = Data)

Residuals:
   Min     1Q Median     3Q    Max
 -3786  -1948   -190   1923   4144

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)         45755.2      275.8 165.899  < 2e-16 ***
AlgoritmoCFS        -2127.8      390.0  -5.455 1.23e-07 ***
AlgoritmoULE        -2886.4      390.0  -7.400 2.37e-12 ***
AlgoritmoMonotonic  -9686.6      390.0 -24.835  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2136 on 236 degrees of freedom
Multiple R-squared:  0.7458,    Adjusted R-squared:  0.7425
F-statistic: 230.8 on 3 and 236 DF,  p-value: < 2.2e-16
```

A anova: 2 × 4

	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo	3159675063	3	230.7698	6.959249e-70
Residuals	1077095280	236	NA	NA

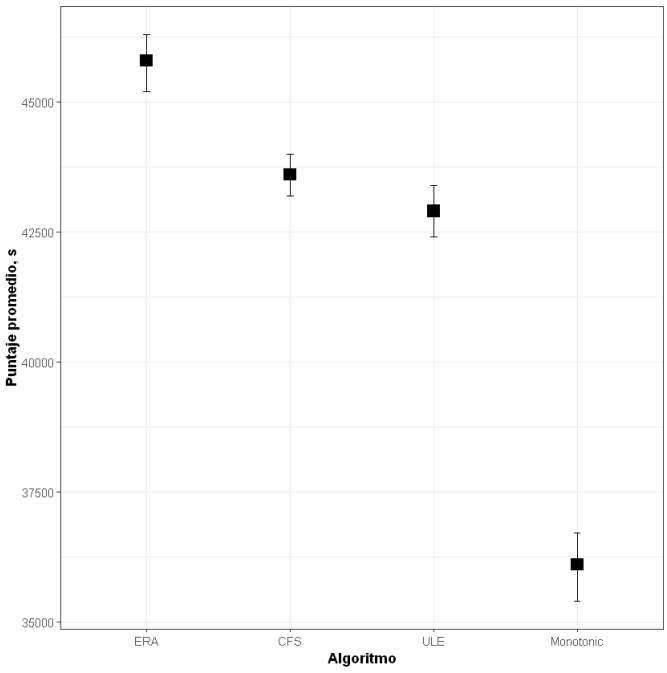
```
 contrast        estimate  SE  df t.ratio p.value
 ERA - CFS           2128 390 236   5.455  <.0001
 ERA - ULE           2886 390 236   7.400  <.0001
 ERA - Monotonic     9687 390 236  24.835  <.0001
 CFS - ULE            759 390 236   1.945  0.2122
 CFS - Monotonic     7559 390 236  19.380  <.0001
 ULE - Monotonic     6800 390 236  17.435  <.0001

P value adjustment: tukey method for comparing a family of 4 estimates
```

```
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 4 × 7

	Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
4	Monotonic	36068.55	275.8007	236	35376.27	36760.83	a
3	ULE	42868.75	275.8007	236	42176.47	43561.03	b
2	CFS	43627.42	275.8007	236	42935.14	44319.69	b
1	ERA	45755.20	275.8007	236	45062.92	46447.48	c

```
Warning message in y + params$y:
"longer object length is not a multiple of shorter object length"
```



In [ ]:

In [1]:
```r
if(!require(FrF2)){install.packages("FrF2")}

# Donde nfactors es numero de factores y nruns es numero de ejecuciones.
dsg <- FrF2(nfactors = 5, nruns = 8)
# 32 -> 8 experimentos
# 1/4 2^k

summary(dsg) # Diseño con la resolución más alta posible.

# Se van a requerir dos funciones generadoras.

# $generators
# [1] D=AB E=AC

# Numero de ejecuciones.
# Como los factores están solapados.

# Otra manera

# Resolución -> 1/2 2^k | Cuantos factores se consideran todas sus combinaciones.
# Donde nfactors es numero de factores y resolution es numero de factores principales.
dsg <- FrF2(nfactors = 5, resolution = 4)
summary(dsg)

# R solo muestra las interacciones entre factores principales e interacciones de dos niveles
# debido a la propiedad de escasez de efectos. Para interacciones niveles superiores no se analizan.
```

```
Loading required package: FrF2

Warning message:
"package 'FrF2' was built under R version 4.2.3"
Loading required package: DoE.base

Warning message:
"package 'DoE.base' was built under R version 4.2.3"
Loading required package: grid

Loading required package: conf.design

Registered S3 method overwritten by 'DoE.base':
  method           from
  factorize.factor conf.design


Attaching package: 'DoE.base'


The following objects are masked from 'package:stats':

    aov, lm


The following object is masked from 'package:graphics':

    plot.design


The following object is masked from 'package:base':

    lengths
```

```
Call:
FrF2(nfactors = 5, nruns = 8)

Experimental design of type  FrF2
8  runs

Factor settings (scale ends):
   A  B  C  D  E
1 -1 -1 -1 -1 -1
2  1  1  1  1  1

Design generating information:
$legend
[1] A=A B=B C=C D=D E=E

$generators
[1] D=AB E=AC


Alias structure:
$main
[1] A=BD=CE B=AD    C=AE    D=AB    E=AC

$fi2
[1] BC=DE BE=CD


The design itself:
   A  B  C  D  E
1 -1 -1 -1  1  1
2  1 -1  1 -1  1
3 -1  1  1 -1 -1
4  1 -1 -1 -1 -1
5  1  1  1  1  1
6 -1  1 -1 -1  1
7 -1 -1  1  1 -1
8  1  1 -1  1 -1
class=design, type= FrF2
Call:
FrF2(nfactors = 5, resolution = 4)

Experimental design of type  FrF2
16  runs

Factor settings (scale ends):
   A  B  C  D  E
1 -1 -1 -1 -1 -1
2  1  1  1  1  1

Design generating information:
$legend
[1] A=A B=B C=C D=D E=E

$generators
[1] E=ABCD


Alias structure:
[[1]]
[1] no aliasing among main effects and 2fis


The design itself:
    A  B  C  D  E
1   1  1 -1 -1  1
2   1  1  1  1  1
3  -1  1  1  1 -1
4  -1 -1 -1  1 -1
5  -1 -1 -1 -1  1
6  -1 -1  1 -1 -1
7   1 -1 -1  1  1
8   1 -1 -1 -1 -1
9  -1  1 -1  1  1
10 -1  1  1 -1  1
11 -1  1 -1 -1 -1
12  1 -1  1  1 -1
```

```
13  1 -1  1 -1  1
14  1  1  1 -1 -1
15 -1 -1  1  1  1
16  1  1 -1  1 -1
class=design, type= FrF2
```

In [ ]:

In [1]:

```
# ANOVA Monofactorial con bloques.

# 1. Carga inicial de datos.

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


ln <- ("
Algoritmo       Computadora       Tiempo
'Algoritmo A'  'Computadora 1'    12976
'Algoritmo A'  'Computadora 1'    14854
'Algoritmo A'  'Computadora 1'    13627
'Algoritmo A'  'Computadora 1'     9850
'Algoritmo A'  'Computadora 1'    14466
'Algoritmo A'  'Computadora 1'    11598
'Algoritmo A'  'Computadora 1'    13184
'Algoritmo A'  'Computadora 1'    13096
'Algoritmo A'  'Computadora 1'    14895
'Algoritmo A'  'Computadora 1'    15986
'Algoritmo A'  'Computadora 1'    12327
'Algoritmo A'  'Computadora 1'    11168
'Algoritmo A'  'Computadora 1'     9913
'Algoritmo A'  'Computadora 1'    11698
'Algoritmo A'  'Computadora 1'    16033
'Algoritmo A'  'Computadora 1'    13763
'Algoritmo A'  'Computadora 1'    10237
'Algoritmo A'  'Computadora 1'    13208
'Algoritmo A'  'Computadora 1'    15407
'Algoritmo A'  'Computadora 1'    13587
'Algoritmo A'  'Computadora 2'     9033
'Algoritmo A'  'Computadora 2'    11253
'Algoritmo A'  'Computadora 2'    11842
'Algoritmo A'  'Computadora 2'     9018
'Algoritmo A'  'Computadora 2'    11091
'Algoritmo A'  'Computadora 2'    11143
'Algoritmo A'  'Computadora 2'    12429
'Algoritmo A'  'Computadora 2'    12456
'Algoritmo A'  'Computadora 2'    12250
'Algoritmo A'  'Computadora 2'    13449
'Algoritmo A'  'Computadora 2'    11872
'Algoritmo A'  'Computadora 2'    10463
'Algoritmo A'  'Computadora 2'     9311
'Algoritmo A'  'Computadora 2'     9677
'Algoritmo A'  'Computadora 2'    12941
'Algoritmo A'  'Computadora 2'    11260
'Algoritmo A'  'Computadora 2'     9269
'Algoritmo A'  'Computadora 2'    13926
'Algoritmo A'  'Computadora 2'    14670
'Algoritmo A'  'Computadora 2'    11988
'Algoritmo B'  'Computadora 1'    11080
'Algoritmo B'  'Computadora 1'    12089
'Algoritmo B'  'Computadora 1'    12538
'Algoritmo B'  'Computadora 1'    10571
'Algoritmo B'  'Computadora 1'    12010
'Algoritmo B'  'Computadora 1'    12598
'Algoritmo B'  'Computadora 1'    13543
'Algoritmo B'  'Computadora 1'    13547
'Algoritmo B'  'Computadora 1'    13217
'Algoritmo B'  'Computadora 1'    15297
'Algoritmo B'  'Computadora 1'    12210
'Algoritmo B'  'Computadora 1'    11299
'Algoritmo B'  'Computadora 1'    10067
'Algoritmo B'  'Computadora 1'    11279
'Algoritmo B'  'Computadora 1'    14006
'Algoritmo B'  'Computadora 1'    12099
'Algoritmo B'  'Computadora 1'    11581
'Algoritmo B'  'Computadora 1'    14012
'Algoritmo B'  'Computadora 1'    15069
'Algoritmo B'  'Computadora 1'    12000
```

```
'Algoritmo B'  'Computadora 2'    12000
'Algoritmo B'  'Computadora 2'    14011
'Algoritmo B'  'Computadora 2'    13508
'Algoritmo B'  'Computadora 2'     9506
'Algoritmo B'  'Computadora 2'    14005
'Algoritmo B'  'Computadora 2'    11514
'Algoritmo B'  'Computadora 2'    13001
'Algoritmo B'  'Computadora 2'    13220
'Algoritmo B'  'Computadora 2'    14211
'Algoritmo B'  'Computadora 2'    15016
'Algoritmo B'  'Computadora 2'    12504
'Algoritmo B'  'Computadora 2'    11501
'Algoritmo B'  'Computadora 2'     9506
'Algoritmo B'  'Computadora 2'    11514
'Algoritmo B'  'Computadora 2'    16005
'Algoritmo B'  'Computadora 2'    13018
'Algoritmo B'  'Computadora 2'    10503
'Algoritmo B'  'Computadora 2'    13015
'Algoritmo B'  'Computadora 2'    17000
'Algoritmo B'  'Computadora 2'    13020
'Algoritmo C'  'Computadora 1'     9148
'Algoritmo C'  'Computadora 1'    11247
'Algoritmo C'  'Computadora 1'    11571
'Algoritmo C'  'Computadora 1'     9212
'Algoritmo C'  'Computadora 1'    11355
'Algoritmo C'  'Computadora 1'    11848
'Algoritmo C'  'Computadora 1'    12171
'Algoritmo C'  'Computadora 1'    12360
'Algoritmo C'  'Computadora 1'    12053
'Algoritmo C'  'Computadora 1'    13219
'Algoritmo C'  'Computadora 1'    11642
'Algoritmo C'  'Computadora 1'    10918
'Algoritmo C'  'Computadora 1'     9223
'Algoritmo C'  'Computadora 1'     9574
'Algoritmo C'  'Computadora 1'    12245
'Algoritmo C'  'Computadora 1'    11781
'Algoritmo C'  'Computadora 1'     9588
'Algoritmo C'  'Computadora 1'    13093
'Algoritmo C'  'Computadora 1'    14155
'Algoritmo C'  'Computadora 1'    11309
'Algoritmo C'  'Computadora 2'    12511
'Algoritmo C'  'Computadora 2'    14375
'Algoritmo C'  'Computadora 2'    13546
'Algoritmo C'  'Computadora 2'     9962
'Algoritmo C'  'Computadora 2'    14273
'Algoritmo C'  'Computadora 2'    11515
'Algoritmo C'  'Computadora 2'    13556
'Algoritmo C'  'Computadora 2'    13121
'Algoritmo C'  'Computadora 2'    14205
'Algoritmo C'  'Computadora 2'    15424
'Algoritmo C'  'Computadora 2'    12778
'Algoritmo C'  'Computadora 2'    11096
'Algoritmo C'  'Computadora 2'     9364
'Algoritmo C'  'Computadora 2'    11521
'Algoritmo C'  'Computadora 2'    16367
'Algoritmo C'  'Computadora 2'    13060
'Algoritmo C'  'Computadora 2'    10991
'Algoritmo C'  'Computadora 2'    13048
'Algoritmo C'  'Computadora 2'    15078
'Algoritmo C'  'Computadora 2'    13443"
)

# Se introduce la tabla.
Data <- read.table(textConnection(ln), header=TRUE)

# Se ordenan los datos según los ingresamos. (Evitar orden alfabético por R).
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))
Data$Computadora <- factor(Data$Computadora, levels = unique(Data$Computadora))

# 2. Verificación de la lectura de datos.

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(ln)
```

```r
# 3. Resumen organizado.

# Se agrega "+ Computadora" para que la tabla aparezca como en clase.
Summarize(Tiempo ~ Algoritmo + Computadora, data = Data, digits = 3)

# 4. Diagrama de cajas

M <- tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

boxplot(Tiempo ~ Algoritmo + Computadora, data = Data)

# 5. Información de promedios e intervalos de confianza.

Sum <- groupwiseMean(Tiempo ~ Algoritmo, data = Data, conf = 0.95, digits = 3, traditional = FALSE, percenti
Sum

# 6. Gráficos de promedios e intervalos de confianza.

library(ggplot2)
ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Tiempo promedio, s")

# 6.1 Información de promedios e intervalos de confianza, cambio para considerar la computadora.

Sum <- groupwiseMean(Tiempo ~ Algoritmo + Computadora,
                     data = Data, conf = 0.95,
                     digits = 3, traditional = FALSE,
                     percentile = TRUE)
Sum

ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Tiempo promedio, s")

# 7. Modelo lineal.

model <- lm(Tiempo ~ Algoritmo + Computadora, data = Data)
summary(model)

# 8. ANOVA.

library(car)
Anova(model, type = "II")

# 9. Histograma de residuos.

x <- residuals(model)
library(rcompanion)

plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)

# 10. Análisis post-hoc

library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Algoritmo)
```

```r
pairs(marginal, adjust="tukey")

# Funcion cld

library(multcomp)
CLD <- cld(marginal, alpha = 0.05, Letters = letters, adjust = "tukey")
CLD

# Gráfico promedios, intervalos de confianza y letras de separación
CLD$Algoritmo <- factor(CLD$Algoritmo, levels = c("Algoritmo A", "Algoritmo B", "Algoritmo C"))
CLD$.group <- gsub(" ", "", CLD$.group)

library(ggplot2)
ggplot(CLD,
       aes( x = Algoritmo,
            y = lsmean,
            label = .group)) +
       geom_point(shape = 15, size = 4) +
       geom_errorbar(aes(ymin = lower.CL,
                         ymax = upper.CL),
                         width = 0.2,
                         size = 0.7) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold"),
             axis.text = element_text(face = "bold"),
             plot.caption = element_text(hjust = 0)) +

       ylab("Promedio del minimo cuadrado \n
             Tiempo de ejecucion") +

       geom_text(nudge_x = c(0,0,0),
                 nudge_y = c(1100, 1100, 1100),
                 color = "black")
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

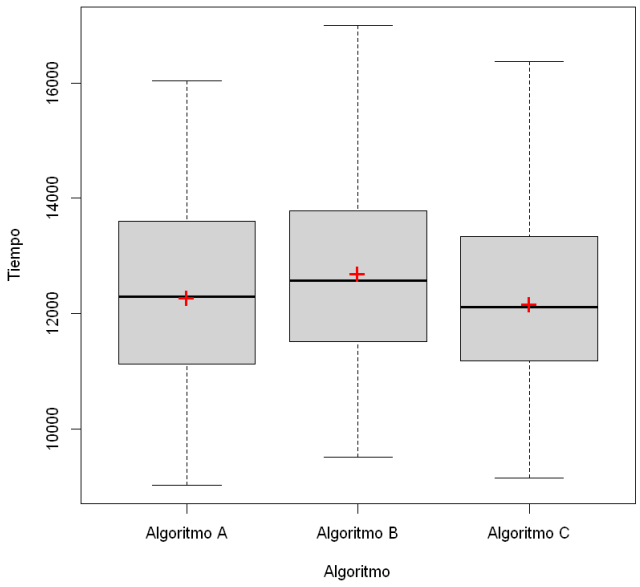
A data.frame: 9 × 3

	Algoritmo	Computadora	Tiempo
	<fct>	<fct>	<chr>
1	Algoritmo A	Computadora 1	12976
2	Algoritmo A	Computadora 1	14854
3	Algoritmo A	Computadora 1	13627
4	Algoritmo A	Computadora 1	9850
...	NA	NA	...
117	Algoritmo C	Computadora 2	10991
118	Algoritmo C	Computadora 2	13048
119	Algoritmo C	Computadora 2	15078
120	Algoritmo C	Computadora 2	13443

```
'data.frame':   120 obs. of  3 variables:
 $ Algoritmo  : Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Computadora: Factor w/ 2 levels "Computadora 1",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Tiempo     : int  12976 14854 13627 9850 14466 11598 13184 13096 14895 15986 ...
      Algoritmo         Computadora     Tiempo
 Algoritmo A:40   Computadora 1:60   Min.   : 9018
 Algoritmo B:40   Computadora 2:60   1st Qu.:11258
 Algoritmo C:40                     Median :12288
                                    Mean   :12382
                                    3rd Qu.:13546
                                    Max.   :17000
```

A data.frame: 6 × 10

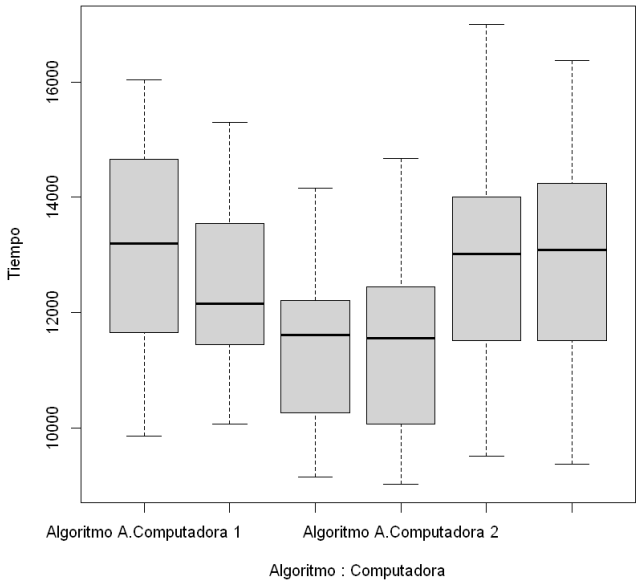
Algoritmo	Computadora	n	mean	sd	min	Q1	median	Q3	max
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	Computadora 1	20	13093.65	1903.929	9850	11673.0	13196.0	14563.00	16033
Algoritmo B	Computadora 1	20	12505.60	1414.667	10067	11510.5	12154.5	13544.00	15297
Algoritmo C	Computadora 1	20	11385.60	1420.394	9148	10585.5	11606.5	12189.50	14155
Algoritmo A	Computadora 2	20	11467.05	1645.540	9018	10266.5	11551.0	12435.75	14670
Algoritmo B	Computadora 2	20	12878.90	1935.371	9506	11514.0	13016.5	14006.50	17000
Algoritmo C	Computadora 2	20	12961.70	1807.597	9364	11519.5	13090.5	14222.00	16367



A data.frame: 3 × 6

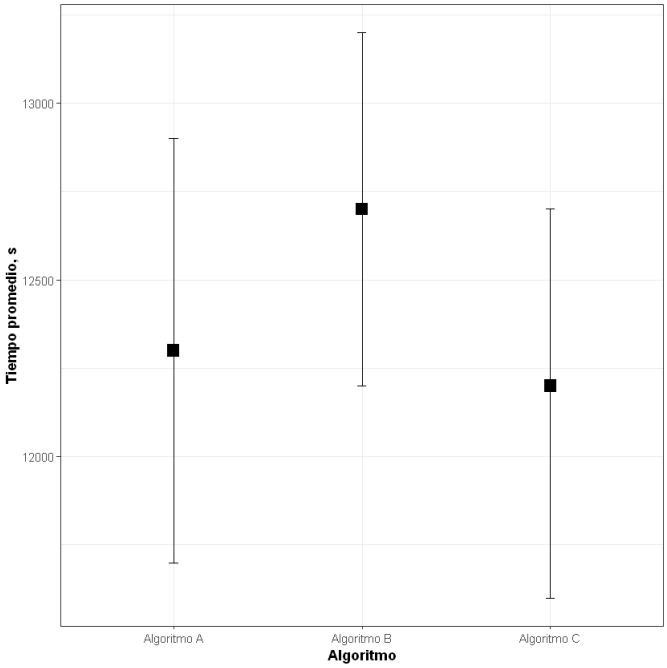
Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	40	12300	0.95	11700	12900
Algoritmo B	40	12700	0.95	12200	13200
Algoritmo C	40	12200	0.95	11600	12700

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```



A data.frame: 6 × 7

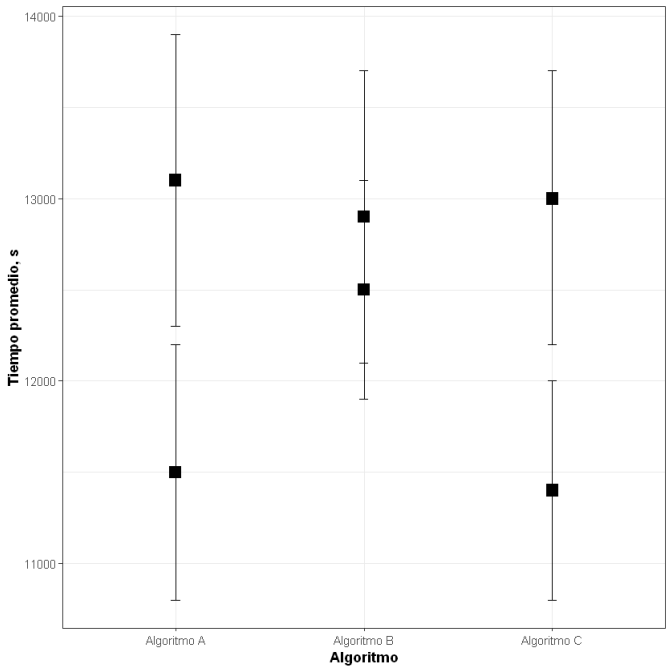
Algoritmo	Computadora	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	Computadora 1	20	13100	0.95	12300	13900
Algoritmo A	Computadora 2	20	11500	0.95	10800	12200
Algoritmo B	Computadora 1	20	12500	0.95	11900	13100
Algoritmo B	Computadora 2	20	12900	0.95	12100	13700
Algoritmo C	Computadora 1	20	11400	0.95	10800	12000
Algoritmo C	Computadora 2	20	13000	0.95	12200	13700

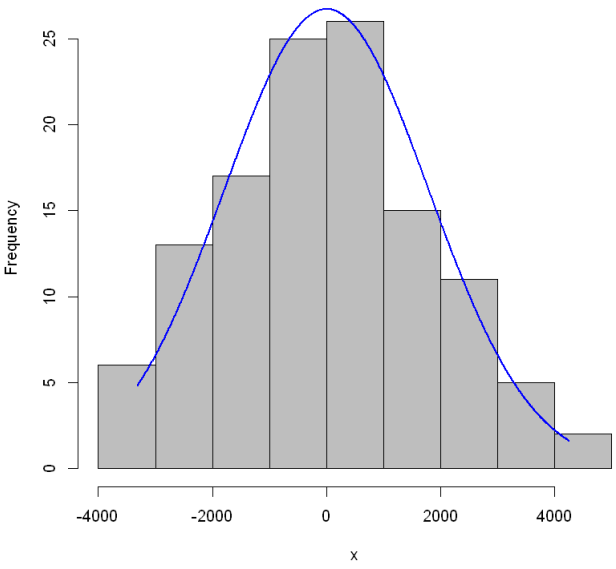


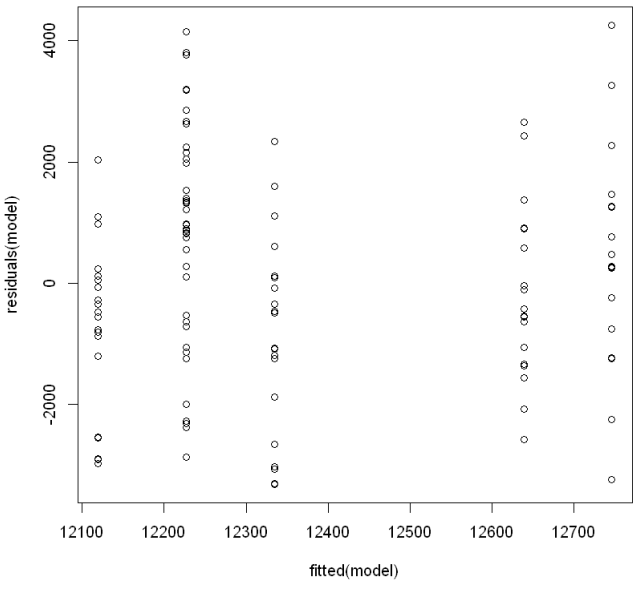

```
Call:
lm(formula = Tiempo ~ Algoritmo + Computadora, data = Data)

Residuals:
    Min      1Q  Median      3Q     Max
-3316.2 -1193.8   -53.6  1226.4  4254.0

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)              12226.5      331.4  36.890   <2e-16 ***
AlgoritmoAlgoritmo B       411.9      405.9   1.015    0.312
AlgoritmoAlgoritmo C      -106.7      405.9  -0.263    0.793
ComputadoraComputadora 2   107.6      331.4   0.325    0.746
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1815 on 116 degrees of freedom
Multiple R-squared:  0.01633,   Adjusted R-squared:  -0.009107
F-statistic: 0.642 on 3 and 116 DF,  p-value: 0.5895
```

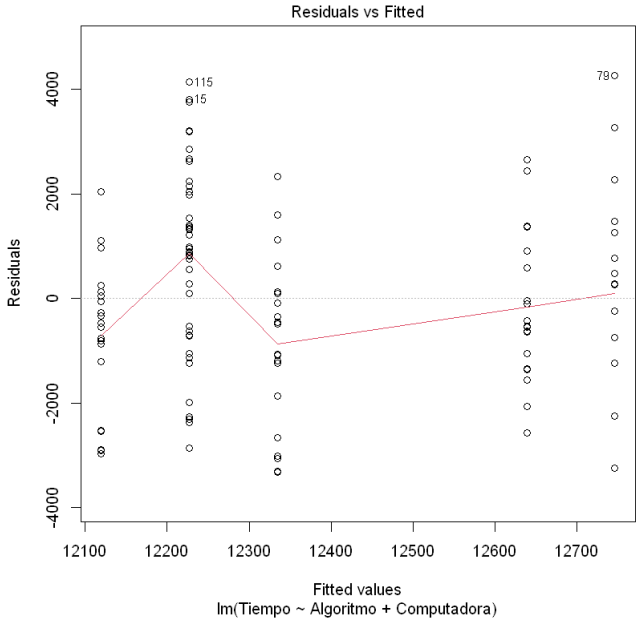
A anova: 3 × 4

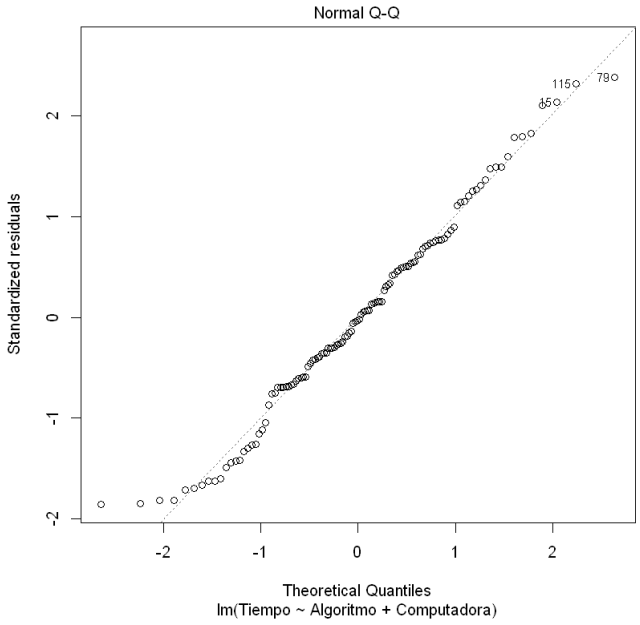
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo	5999899.5	2	0.9103285	0.4052470
Computadora	347332.8	1	0.1053974	0.7460309
Residuals	382273164.9	116	NA	NA

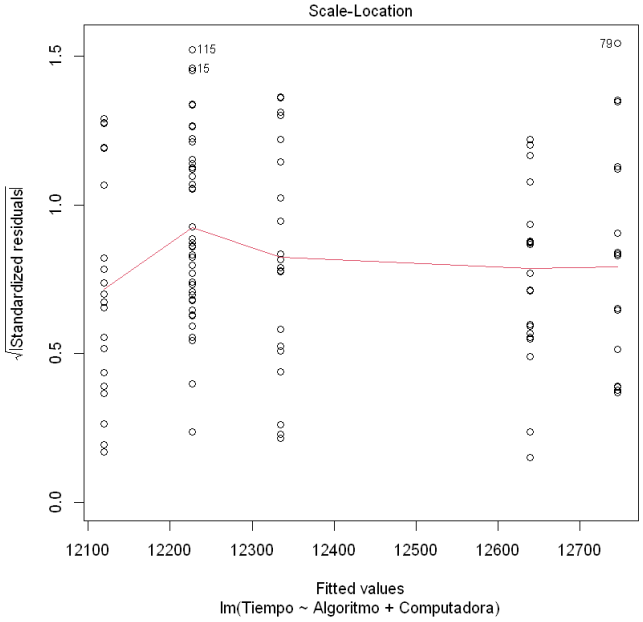

14000
13000
12000
11000
Tiempo promedio, s
Algoritmo A
Algoritmo B
Algoritmo C
Algoritmo


Frequency
0
5
10
15
20
25
-4000
-2000
0
2000
4000
x


residuals(model)
-2000
0
2000
4000
12100
12200
12300
12400
12500
12600
12700
fitted(model)


Residuals vs Fitted
Residuals
Fitted values
lm(Tiempo ~ Algoritmo + Computadora)
115
15
79


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Tiempo ~ Algoritmo + Computadora)
115
15
79

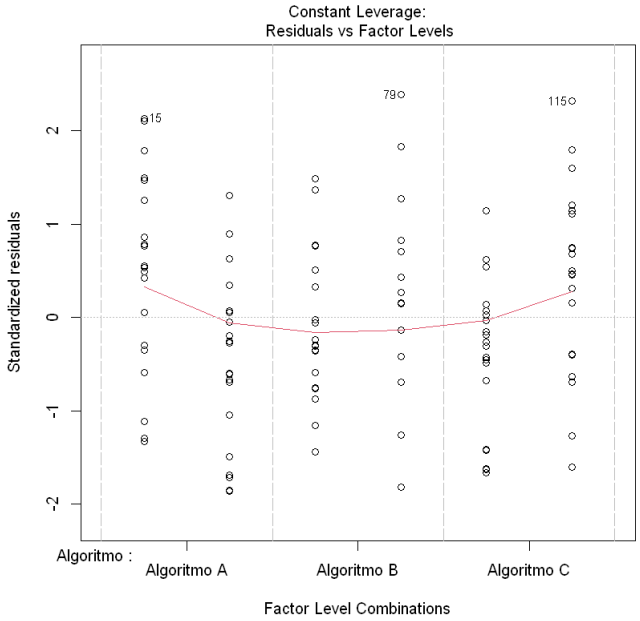

Scale-Location
√|Standardized residuals|
Fitted values
lm(Tiempo ~ Algoritmo + Computadora)
115
15
79

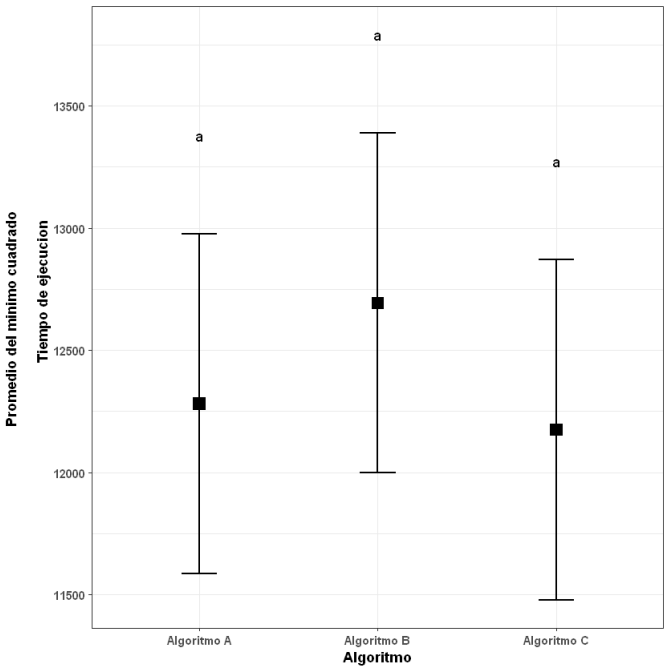
```
 contrast                 estimate  SE  df t.ratio p.value
 Algoritmo A - Algoritmo B     -412 406 116  -1.015  0.5691
 Algoritmo A - Algoritmo C      107 406 116   0.263  0.9626
 Algoritmo B - Algoritmo C      519 406 116   1.278  0.4107

Results are averaged over the levels of: Computadora
P value adjustment: tukey method for comparing a family of 3 estimates
```

```
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 3 × 7

	Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	Algoritmo C	12173.65	287.0304	116	11478.26	12869.04	a
1	Algoritmo A	12280.35	287.0304	116	11584.96	12975.74	a
2	Algoritmo B	12692.25	287.0304	116	11996.86	13387.64	a




a
a
a
13500
13000
12500
12000
11500
Promedio del minimo cuadrado
Tiempo de ejecucion
Algoritmo A
Algoritmo B
Algoritmo C
Algoritmo

In [ ]:

In [1]:

```
# ANOVA Monofactorial con bloques.

# 1. Carga inicial de datos.

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


ln <- ("
Algoritmo       Computadora       Tiempo
'Algoritmo A'  'Computadora 1'    12976
'Algoritmo A'  'Computadora 1'    14854
'Algoritmo A'  'Computadora 1'    13627
'Algoritmo A'  'Computadora 1'     9850
'Algoritmo A'  'Computadora 1'    14466
'Algoritmo A'  'Computadora 1'    11598
'Algoritmo A'  'Computadora 1'    13184
'Algoritmo A'  'Computadora 1'    13096
'Algoritmo A'  'Computadora 1'    14895
'Algoritmo A'  'Computadora 1'    15986
'Algoritmo A'  'Computadora 1'    12327
'Algoritmo A'  'Computadora 1'    11168
'Algoritmo A'  'Computadora 1'     9913
'Algoritmo A'  'Computadora 1'    11698
'Algoritmo A'  'Computadora 1'    16033
'Algoritmo A'  'Computadora 1'    13763
'Algoritmo A'  'Computadora 1'    10237
'Algoritmo A'  'Computadora 1'    13208
'Algoritmo A'  'Computadora 1'    15407
'Algoritmo A'  'Computadora 1'    13587
'Algoritmo A'  'Computadora 2'     9033
'Algoritmo A'  'Computadora 2'    11253
'Algoritmo A'  'Computadora 2'    11842
'Algoritmo A'  'Computadora 2'     9018
'Algoritmo A'  'Computadora 2'    11091
'Algoritmo A'  'Computadora 2'    11143
'Algoritmo A'  'Computadora 2'    12429
'Algoritmo A'  'Computadora 2'    12456
'Algoritmo A'  'Computadora 2'    12250
'Algoritmo A'  'Computadora 2'    13449
'Algoritmo A'  'Computadora 2'    11872
'Algoritmo A'  'Computadora 2'    10463
'Algoritmo A'  'Computadora 2'     9311
'Algoritmo A'  'Computadora 2'     9677
'Algoritmo A'  'Computadora 2'    12941
'Algoritmo A'  'Computadora 2'    11260
'Algoritmo A'  'Computadora 2'     9269
'Algoritmo A'  'Computadora 2'    13926
'Algoritmo A'  'Computadora 2'    14670
'Algoritmo A'  'Computadora 2'    11988
'Algoritmo B'  'Computadora 1'    11080
'Algoritmo B'  'Computadora 1'    12089
'Algoritmo B'  'Computadora 1'    12538
'Algoritmo B'  'Computadora 1'    10571
'Algoritmo B'  'Computadora 1'    12010
'Algoritmo B'  'Computadora 1'    12598
'Algoritmo B'  'Computadora 1'    13543
'Algoritmo B'  'Computadora 1'    13547
'Algoritmo B'  'Computadora 1'    13217
'Algoritmo B'  'Computadora 1'    15297
'Algoritmo B'  'Computadora 1'    12210
'Algoritmo B'  'Computadora 1'    11299
'Algoritmo B'  'Computadora 1'    10067
'Algoritmo B'  'Computadora 1'    11279
'Algoritmo B'  'Computadora 1'    14006
'Algoritmo B'  'Computadora 1'    12099
'Algoritmo B'  'Computadora 1'    11581
'Algoritmo B'  'Computadora 1'    14012
'Algoritmo B'  'Computadora 1'    15069
'Algoritmo B'  'Computadora 1'    12000
```

```
'Algoritmo B'  'Computadora 2'    12000
'Algoritmo B'  'Computadora 2'    14011
'Algoritmo B'  'Computadora 2'    13508
'Algoritmo B'  'Computadora 2'     9506
'Algoritmo B'  'Computadora 2'    14005
'Algoritmo B'  'Computadora 2'    11514
'Algoritmo B'  'Computadora 2'    13001
'Algoritmo B'  'Computadora 2'    13220
'Algoritmo B'  'Computadora 2'    14211
'Algoritmo B'  'Computadora 2'    15016
'Algoritmo B'  'Computadora 2'    12504
'Algoritmo B'  'Computadora 2'    11501
'Algoritmo B'  'Computadora 2'     9506
'Algoritmo B'  'Computadora 2'    11514
'Algoritmo B'  'Computadora 2'    16005
'Algoritmo B'  'Computadora 2'    13018
'Algoritmo B'  'Computadora 2'    10503
'Algoritmo B'  'Computadora 2'    13015
'Algoritmo B'  'Computadora 2'    17000
'Algoritmo B'  'Computadora 2'    13020
'Algoritmo C'  'Computadora 1'     9148
'Algoritmo C'  'Computadora 1'    11247
'Algoritmo C'  'Computadora 1'    11571
'Algoritmo C'  'Computadora 1'     9212
'Algoritmo C'  'Computadora 1'    11355
'Algoritmo C'  'Computadora 1'    11848
'Algoritmo C'  'Computadora 1'    12171
'Algoritmo C'  'Computadora 1'    12360
'Algoritmo C'  'Computadora 1'    12053
'Algoritmo C'  'Computadora 1'    13219
'Algoritmo C'  'Computadora 1'    11642
'Algoritmo C'  'Computadora 1'    10918
'Algoritmo C'  'Computadora 1'     9223
'Algoritmo C'  'Computadora 1'     9574
'Algoritmo C'  'Computadora 1'    12245
'Algoritmo C'  'Computadora 1'    11781
'Algoritmo C'  'Computadora 1'     9588
'Algoritmo C'  'Computadora 1'    13093
'Algoritmo C'  'Computadora 1'    14155
'Algoritmo C'  'Computadora 1'    11309
'Algoritmo C'  'Computadora 2'    12511
'Algoritmo C'  'Computadora 2'    14375
'Algoritmo C'  'Computadora 2'    13546
'Algoritmo C'  'Computadora 2'     9962
'Algoritmo C'  'Computadora 2'    14273
'Algoritmo C'  'Computadora 2'    11515
'Algoritmo C'  'Computadora 2'    13556
'Algoritmo C'  'Computadora 2'    13121
'Algoritmo C'  'Computadora 2'    14205
'Algoritmo C'  'Computadora 2'    15424
'Algoritmo C'  'Computadora 2'    12778
'Algoritmo C'  'Computadora 2'    11096
'Algoritmo C'  'Computadora 2'     9364
'Algoritmo C'  'Computadora 2'    11521
'Algoritmo C'  'Computadora 2'    16367
'Algoritmo C'  'Computadora 2'    13060
'Algoritmo C'  'Computadora 2'    10991
'Algoritmo C'  'Computadora 2'    13048
'Algoritmo C'  'Computadora 2'    15078
'Algoritmo C'  'Computadora 2'    13443"
)

# Se introduce la tabla.
Data <- read.table(textConnection(ln), header=TRUE)

# Se ordenan los datos según los ingresamos. (Evitar orden alfabético por R).
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))
Data$Computadora <- factor(Data$Computadora, levels = unique(Data$Computadora))

# Se elimina Computadora 1 de los datos.

Data <- Data[Data$Computadora != "Computadora 1", ]

# 2. Verificación de la lectura de datos.

library(psych)
```

```r
headTail(Data)
str(Data)
summary(Data)
rm(ln)

# 3. Resumen organizado.

# Se agrega para que la tabla aparezca como en clase.
Summarize(Tiempo ~ Algoritmo, data = Data, digits = 3)

# 4. Diagrama de cajas

M <- tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

# Comentario: Al traslaparse las cajas no se puede concluir aún que haya diferencia estadística entre los gr
# Por ahora se puede notar que la caja para el algoritmo A está ligeramente más abajo que las otras dos.

# 5. Información de promedios e intervalos de confianza.

Sum <- groupwiseMean(Tiempo ~ Algoritmo, data = Data, conf = 0.95, digits = 3, traditional = FALSE, percenti
Sum

# 6. Gráficos de promedios e intervalos de confianza.

library(ggplot2)
ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                  geom_point(shape = 15,
                             size = 4) +
                  theme_bw() +
                  theme(axis.title = element_text(face = "bold")) +
                  ylab("Tiempo promedio, s")

# Al observar los intervalos de confianza, se resalta aún más la diferencia entre el Algoritmo A con respect
# otros dos, pero aún así, su intervalo de confianza se traslapa levemente, por lo que se procede a realizar
# la prueba estadística.

# 7. Modelo lineal.

model <- lm(Tiempo ~ Algoritmo, data = Data)
summary(model)

# There is a statistically significant relationship between the predictor variable and the response variable
# El p-value en este caso es bajo 0.01736, lo que sugiere los factores impactan la variable de respuesta.

# 8. ANOVA.

library(car)
Anova(model, type = "II")

# La prueba ANOVA pasa con un nivel de significancia de 0.05. Se procede a realizar la prueba post-hoc.

# 9. Histograma de residuos.

x <- residuals(model)
library(rcompanion)
plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)

# Datos presentan normalidad y homocedastisidad.

# 10. Análisis post-hoc

library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Algoritmo)
pairs(marginal, adjust="tukey")

# Funcion cld
```

```r
library(multcomp)
CLD <- cld(marginal, alpha = 0.05, Letters = letters, adjust = "tukey")
CLD

# El análisis post-hoc indica que el Algoritmo A es estadísticamente distinto a los algoritmos B y C.

# Gráfico promedios, intervalos de confianza y letras de separación
CLD$Algoritmo <- factor(CLD$Algoritmo, levels = c("Algoritmo A", "Algoritmo B", "Algoritmo C"))
CLD$.group <- gsub(" ", "", CLD$.group)

svg("final-ic-1.svg")
library(ggplot2)
ggplot(CLD,
       aes( x = Algoritmo,
            y = lsmean,
            label = .group)) +
       geom_point(shape = 15, size = 4) +
       geom_errorbar(aes(ymin = lower.CL,
                         ymax = upper.CL),
                         width = 0.2,
                         size = 0.7) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold"),
             axis.text = element_text(face = "bold"),
             plot.caption = element_text(hjust = 0)) +

       ylab("Promedio del minimo cuadrado \n
             Tiempo de ejecucion") +

       geom_text(nudge_x = c(0,0,0),
                 nudge_y = c(1100, 1100, 1100),
                 color = "black")
dev.off()
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 3

	Algoritmo	Computadora	Tiempo
	<fct>	<fct>	<chr>
21	Algoritmo A	Computadora 2	9033
22	Algoritmo A	Computadora 2	11253
23	Algoritmo A	Computadora 2	11842
24	Algoritmo A	Computadora 2	9018
...	NA	NA	...
117	Algoritmo C	Computadora 2	10991
118	Algoritmo C	Computadora 2	13048
119	Algoritmo C	Computadora 2	15078
120	Algoritmo C	Computadora 2	13443

```
'data.frame':   60 obs. of  3 variables:
 $ Algoritmo  : Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Computadora: Factor w/ 2 levels "Computadora 1",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ Tiempo     : int  9033 11253 11842 9018 11091 11143 12429 12456 12250 13449 ...
      Algoritmo         Computadora     Tiempo
 Algoritmo A:20   Computadora 1: 0   Min.   : 9018
 Algoritmo B:20   Computadora 2:60   1st Qu.:11226
 Algoritmo C:20                     Median :12508
                                    Mean   :12436
                                    3rd Qu.:13548
                                    Max.   :17000
```

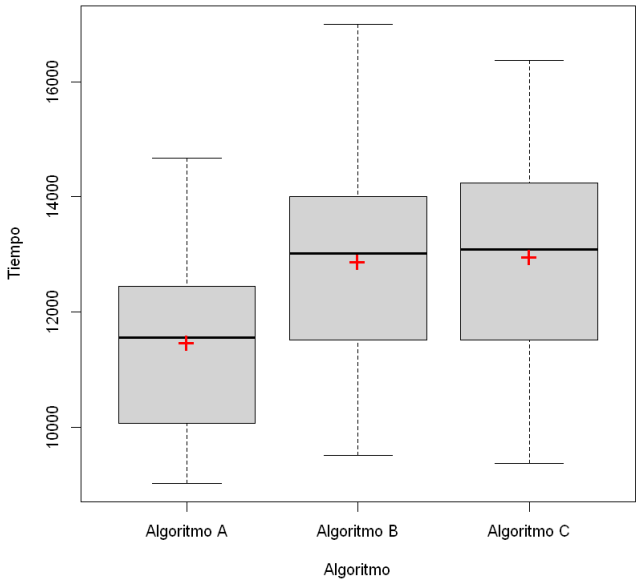
A data.frame: 3 × 9

Algoritmo	n	mean	sd	min	Q1	median	Q3	max
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	11467.05	1645.540	9018	10266.5	11551.0	12435.75	14670
Algoritmo B	20	12878.90	1935.371	9506	11514.0	13016.5	14006.50	17000
Algoritmo C	20	12961.70	1807.597	9364	11519.5	13090.5	14222.00	16367

A data.frame: 3 × 6

Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	11500	0.95	10800	12200
Algoritmo B	20	12900	0.95	12100	13700
Algoritmo C	20	13000	0.95	12200	13700

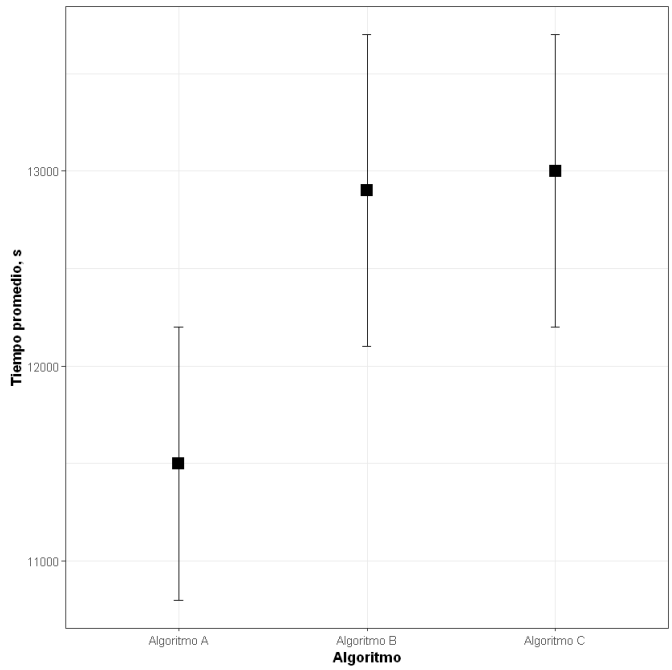
```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```

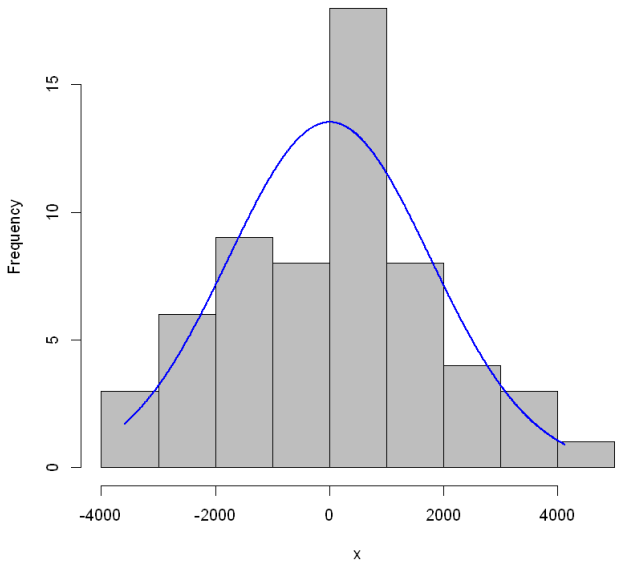


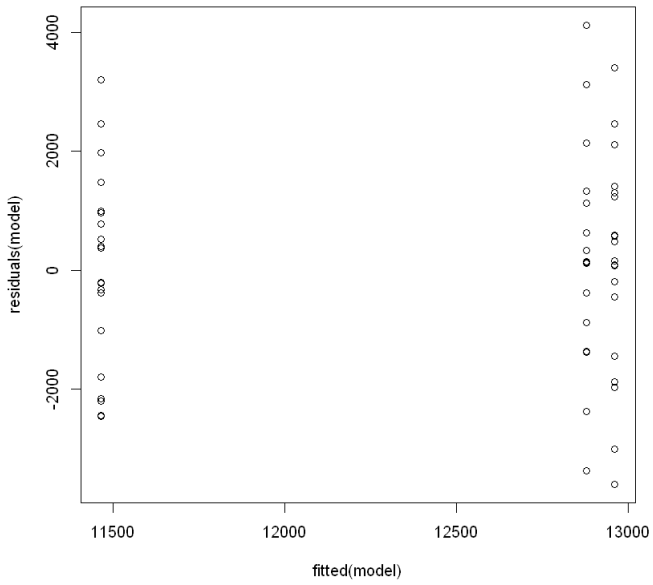

```
Call:
lm(formula = Tiempo ~ Algoritmo, data = Data)

Residuals:
    Min      1Q  Median      3Q     Max
-3597.7 -1368.2   137.6  1127.6  4121.1

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           11467.0      402.5  28.489   <2e-16 ***
AlgoritmoAlgoritmo B   1411.9      569.2   2.480   0.0161 *
AlgoritmoAlgoritmo C   1494.7      569.2   2.626   0.0111 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1800 on 57 degrees of freedom
Multiple R-squared:  0.1326,    Adjusted R-squared:  0.1021
F-statistic: 4.356 on 2 and 57 DF,  p-value: 0.01736
```

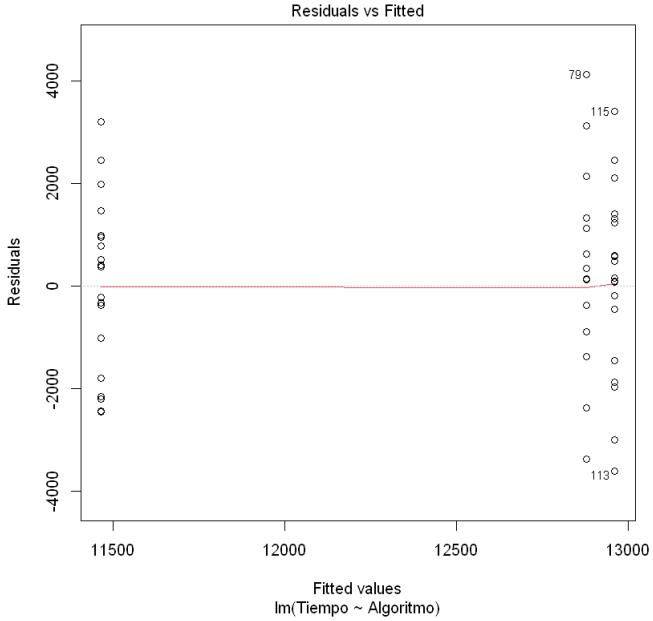
A anova: 2 × 4

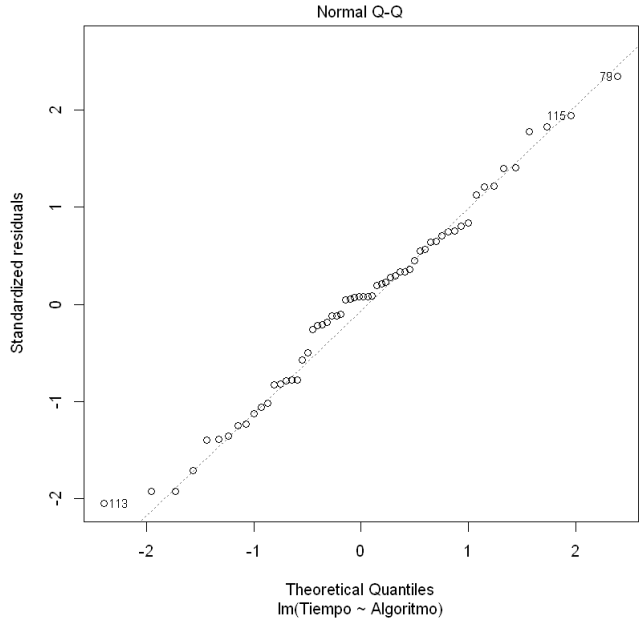
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo	28227699	2	4.355736	0.01736437
Residuals	184696537	57	NA	NA

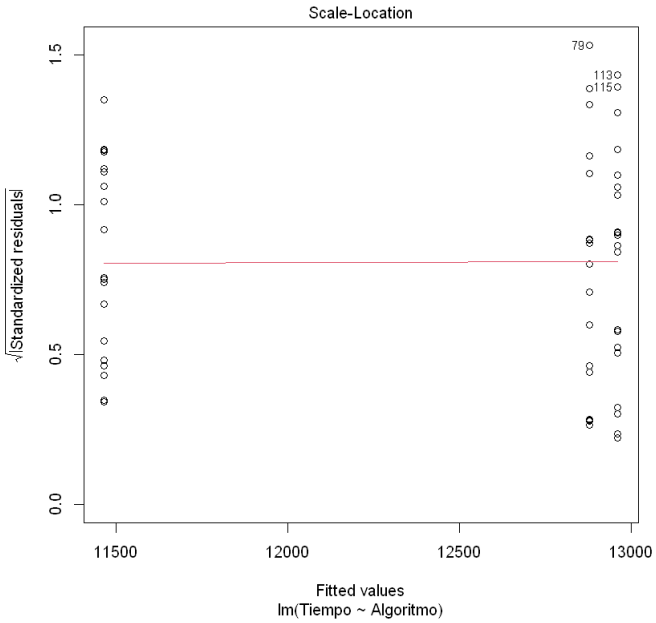



Frequency
0
5
10
15
-4000
-2000
0
2000
4000
x


residuals(model)
-2000
0
2000
4000
11500
12000
12500
13000
fitted(model)


Residuals vs Fitted
Residuals
-4000
-2000
0
2000
4000
79
115
113
11500
12000
12500
13000
Fitted values
lm(Tiempo ~ Algoritmo)





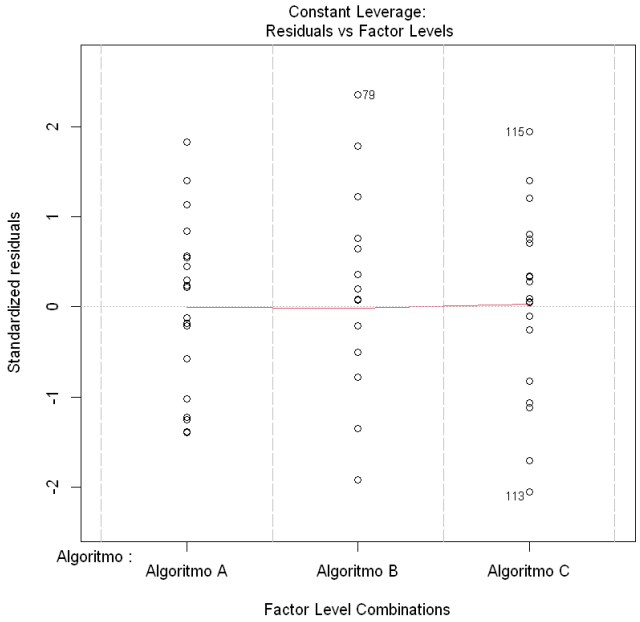

```
 contrast                  estimate  SE df t.ratio p.value
 Algoritmo A - Algoritmo B  -1411.8 569 57  -2.480  0.0420
 Algoritmo A - Algoritmo C  -1494.7 569 57  -2.626  0.0294
 Algoritmo B - Algoritmo C    -82.8 569 57  -0.145  0.9884

P value adjustment: tukey method for comparing a family of 3 estimates
```

```
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 3 × 7

	Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1	Algoritmo A	11467.05	402.5103	57	10476.90	12457.20	a
2	Algoritmo B	12878.90	402.5103	57	11888.75	13869.05	b
3	Algoritmo C	12961.70	402.5103	57	11971.55	13951.85	b

png: 2



In [ ]:

In [1]:

```
# ANOVA Monofactorial con bloques.

# 1. Carga inicial de datos.

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


ln <- ("
Algoritmo       Computadora       Tiempo
'Algoritmo A'  'Computadora 1'    12976
'Algoritmo A'  'Computadora 1'    14854
'Algoritmo A'  'Computadora 1'    13627
'Algoritmo A'  'Computadora 1'     9850
'Algoritmo A'  'Computadora 1'    14466
'Algoritmo A'  'Computadora 1'    11598
'Algoritmo A'  'Computadora 1'    13184
'Algoritmo A'  'Computadora 1'    13096
'Algoritmo A'  'Computadora 1'    14895
'Algoritmo A'  'Computadora 1'    15986
'Algoritmo A'  'Computadora 1'    12327
'Algoritmo A'  'Computadora 1'    11168
'Algoritmo A'  'Computadora 1'     9913
'Algoritmo A'  'Computadora 1'    11698
'Algoritmo A'  'Computadora 1'    16033
'Algoritmo A'  'Computadora 1'    13763
'Algoritmo A'  'Computadora 1'    10237
'Algoritmo A'  'Computadora 1'    13208
'Algoritmo A'  'Computadora 1'    15407
'Algoritmo A'  'Computadora 1'    13587
'Algoritmo A'  'Computadora 2'     9033
'Algoritmo A'  'Computadora 2'    11253
'Algoritmo A'  'Computadora 2'    11842
'Algoritmo A'  'Computadora 2'     9018
'Algoritmo A'  'Computadora 2'    11091
'Algoritmo A'  'Computadora 2'    11143
'Algoritmo A'  'Computadora 2'    12429
'Algoritmo A'  'Computadora 2'    12456
'Algoritmo A'  'Computadora 2'    12250
'Algoritmo A'  'Computadora 2'    13449
'Algoritmo A'  'Computadora 2'    11872
'Algoritmo A'  'Computadora 2'    10463
'Algoritmo A'  'Computadora 2'     9311
'Algoritmo A'  'Computadora 2'     9677
'Algoritmo A'  'Computadora 2'    12941
'Algoritmo A'  'Computadora 2'    11260
'Algoritmo A'  'Computadora 2'     9269
'Algoritmo A'  'Computadora 2'    13926
'Algoritmo A'  'Computadora 2'    14670
'Algoritmo A'  'Computadora 2'    11988
'Algoritmo B'  'Computadora 1'    11080
'Algoritmo B'  'Computadora 1'    12089
'Algoritmo B'  'Computadora 1'    12538
'Algoritmo B'  'Computadora 1'    10571
'Algoritmo B'  'Computadora 1'    12010
'Algoritmo B'  'Computadora 1'    12598
'Algoritmo B'  'Computadora 1'    13543
'Algoritmo B'  'Computadora 1'    13547
'Algoritmo B'  'Computadora 1'    13217
'Algoritmo B'  'Computadora 1'    15297
'Algoritmo B'  'Computadora 1'    12210
'Algoritmo B'  'Computadora 1'    11299
'Algoritmo B'  'Computadora 1'    10067
'Algoritmo B'  'Computadora 1'    11279
'Algoritmo B'  'Computadora 1'    14006
'Algoritmo B'  'Computadora 1'    12099
'Algoritmo B'  'Computadora 1'    11581
'Algoritmo B'  'Computadora 1'    14012
'Algoritmo B'  'Computadora 1'    15069
'Algoritmo B'  'Computadora 1'    12000
```

```
'Algoritmo B'  'Computadora 2'    12000
'Algoritmo B'  'Computadora 2'    14011
'Algoritmo B'  'Computadora 2'    13508
'Algoritmo B'  'Computadora 2'     9506
'Algoritmo B'  'Computadora 2'    14005
'Algoritmo B'  'Computadora 2'    11514
'Algoritmo B'  'Computadora 2'    13001
'Algoritmo B'  'Computadora 2'    13220
'Algoritmo B'  'Computadora 2'    14211
'Algoritmo B'  'Computadora 2'    15016
'Algoritmo B'  'Computadora 2'    12504
'Algoritmo B'  'Computadora 2'    11501
'Algoritmo B'  'Computadora 2'     9506
'Algoritmo B'  'Computadora 2'    11514
'Algoritmo B'  'Computadora 2'    16005
'Algoritmo B'  'Computadora 2'    13018
'Algoritmo B'  'Computadora 2'    10503
'Algoritmo B'  'Computadora 2'    13015
'Algoritmo B'  'Computadora 2'    17000
'Algoritmo B'  'Computadora 2'    13020
'Algoritmo C'  'Computadora 1'     9148
'Algoritmo C'  'Computadora 1'    11247
'Algoritmo C'  'Computadora 1'    11571
'Algoritmo C'  'Computadora 1'     9212
'Algoritmo C'  'Computadora 1'    11355
'Algoritmo C'  'Computadora 1'    11848
'Algoritmo C'  'Computadora 1'    12171
'Algoritmo C'  'Computadora 1'    12360
'Algoritmo C'  'Computadora 1'    12053
'Algoritmo C'  'Computadora 1'    13219
'Algoritmo C'  'Computadora 1'    11642
'Algoritmo C'  'Computadora 1'    10918
'Algoritmo C'  'Computadora 1'     9223
'Algoritmo C'  'Computadora 1'     9574
'Algoritmo C'  'Computadora 1'    12245
'Algoritmo C'  'Computadora 1'    11781
'Algoritmo C'  'Computadora 1'     9588
'Algoritmo C'  'Computadora 1'    13093
'Algoritmo C'  'Computadora 1'    14155
'Algoritmo C'  'Computadora 1'    11309
'Algoritmo C'  'Computadora 2'    12511
'Algoritmo C'  'Computadora 2'    14375
'Algoritmo C'  'Computadora 2'    13546
'Algoritmo C'  'Computadora 2'     9962
'Algoritmo C'  'Computadora 2'    14273
'Algoritmo C'  'Computadora 2'    11515
'Algoritmo C'  'Computadora 2'    13556
'Algoritmo C'  'Computadora 2'    13121
'Algoritmo C'  'Computadora 2'    14205
'Algoritmo C'  'Computadora 2'    15424
'Algoritmo C'  'Computadora 2'    12778
'Algoritmo C'  'Computadora 2'    11096
'Algoritmo C'  'Computadora 2'     9364
'Algoritmo C'  'Computadora 2'    11521
'Algoritmo C'  'Computadora 2'    16367
'Algoritmo C'  'Computadora 2'    13060
'Algoritmo C'  'Computadora 2'    10991
'Algoritmo C'  'Computadora 2'    13048
'Algoritmo C'  'Computadora 2'    15078
'Algoritmo C'  'Computadora 2'    13443"
)

# Se introduce la tabla.
Data <- read.table(textConnection(ln), header=TRUE)

# Se ordenan los datos según los ingresamos. (Evitar orden alfabético por R).
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))
Data$Computadora <- factor(Data$Computadora, levels = unique(Data$Computadora))

# 2. Verificación de la lectura de datos.

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(ln)
```

```r
# 3. Resumen organizado.

# Se agrega ": Computadora" para que la tabla aparezca como en clase.
Summarize(Tiempo ~ Algoritmo : Computadora, data = Data, digits = 3)

# 4. Diagrama de cajas

M <- tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

boxplot(Tiempo ~ Algoritmo : Computadora, data = Data)

# 5. Información de promedios e intervalos de confianza.

Sum <- groupwiseMean(Tiempo ~ Algoritmo, data = Data, conf = 0.95, digits = 3, traditional = FALSE, percenti
Sum

# 6. Gráficos de promedios e intervalos de confianza.

library(ggplot2)
ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Tiempo promedio, s")

# 6.1 Información de promedios e intervalos de confianza, cambio para considerar la computadora.

Sum <- groupwiseMean(Tiempo ~ Algoritmo : Computadora,
                     data = Data, conf = 0.95,
                     digits = 3, traditional = FALSE,
                     percentile = TRUE)
Sum

ggplot(Sum,
       aes(x = Algoritmo, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Tiempo promedio, s")

# 7. Modelo lineal.

model <- lm(Tiempo ~ Algoritmo : Computadora, data = Data)
summary(model)

# El p-value en este caso es bajo 0.002001, lo que sugiere los factores impactan la variable de respuesta.

# 8. ANOVA.

library(car)
Anova(model, type = "II")

# La prueba ANOVA pasa con un nivel de significancia de 0.01. Se procede a realizar la prueba post-hoc.

# 9. Histograma de residuos.

x <- residuals(model)
library(rcompanion)
plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)

# Se cumple la normalidad, pero hay un ligero patrón en el gráfico donde se evalua la homocedasticidad.
```

```r
# 10. Análisis post-hoc

library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Algoritmo : Computadora)
pairs(marginal, adjust="tukey")

# Funcion cld

library(multcomp)
CLD <- cld(marginal, alpha = 0.05, Letters = letters, adjust = "tukey")
CLD

# Gráfico promedios, intervalos de confianza y letras de separación
CLD$Algoritmo <- factor(CLD$Algoritmo, levels = c("Algoritmo A", "Algoritmo B", "Algoritmo C"))
CLD$.group <- gsub(" ", "", CLD$.group)

library(ggplot2)
svg("final-ic-3.svg")
ggplot(CLD,
       aes( x = Algoritmo,
            y = lsmean,
            label = .group)) +
       geom_point(shape = 15, size = 4) +
       geom_errorbar(aes(ymin = lower.CL,
                         ymax = upper.CL),
                         width = 0.2,
                         size = 0.7) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold"),
             axis.text = element_text(face = "bold"),
             plot.caption = element_text(hjust = 0)) +

       ylab("Promedio del minimo cuadrado \n
             Tiempo de ejecucion") +

       geom_text(nudge_x = c(0,0,0),
                 nudge_y = c(1100, 1100, 1100),
                 color = "black")
dev.off()

# La prueba de mínimos cuadrados al ser más estricta y en donde se
# reduce la distancia en comparaciones, indica que los grupos
# no son significativamente distintos, esto contrasta con la respuesta
# del ANOVA.

# Sin embargo si se agrega en ": Computadora" en la función lsmeans, este sería el resultado:
#
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

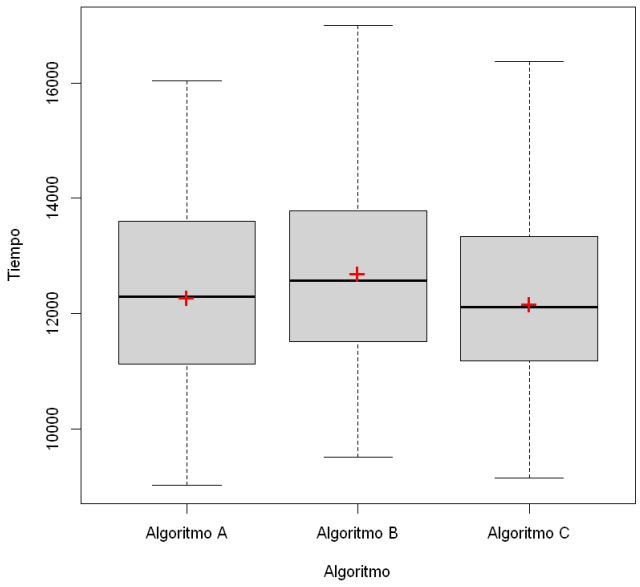
A data.frame: 9 × 3

	Algoritmo	Computadora	Tiempo
	<fct>	<fct>	<chr>
1	Algoritmo A	Computadora 1	12976
2	Algoritmo A	Computadora 1	14854
3	Algoritmo A	Computadora 1	13627
4	Algoritmo A	Computadora 1	9850
...	NA	NA	...
117	Algoritmo C	Computadora 2	10991
118	Algoritmo C	Computadora 2	13048
119	Algoritmo C	Computadora 2	15078
120	Algoritmo C	Computadora 2	13443

```
'data.frame':   120 obs. of  3 variables:
 $ Algoritmo  : Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Computadora: Factor w/ 2 levels "Computadora 1",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Tiempo     : int  12976 14854 13627 9850 14466 11598 13184 13096 14895 15986 ...
      Algoritmo         Computadora     Tiempo
 Algoritmo A:40   Computadora 1:60   Min.   : 9018
 Algoritmo B:40   Computadora 2:60   1st Qu.:11258
 Algoritmo C:40                     Median :12288
                                    Mean   :12382
                                    3rd Qu.:13546
                                    Max.   :17000
```

A data.frame: 6 × 10

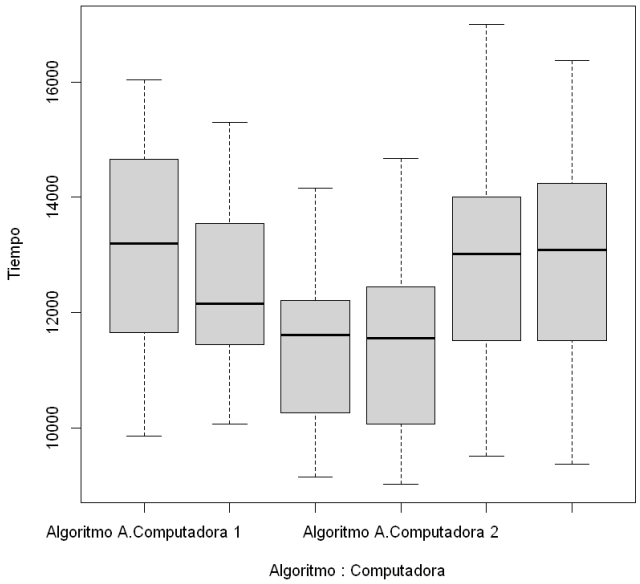
Algoritmo	Computadora	n	mean	sd	min	Q1	median	Q3	max
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	Computadora 1	20	13093.65	1903.929	9850	11673.0	13196.0	14563.00	16033
Algoritmo B	Computadora 1	20	12505.60	1414.667	10067	11510.5	12154.5	13544.00	15297
Algoritmo C	Computadora 1	20	11385.60	1420.394	9148	10585.5	11606.5	12189.50	14155
Algoritmo A	Computadora 2	20	11467.05	1645.540	9018	10266.5	11551.0	12435.75	14670
Algoritmo B	Computadora 2	20	12878.90	1935.371	9506	11514.0	13016.5	14006.50	17000
Algoritmo C	Computadora 2	20	12961.70	1807.597	9364	11519.5	13090.5	14222.00	16367



A data.frame: 3 × 6

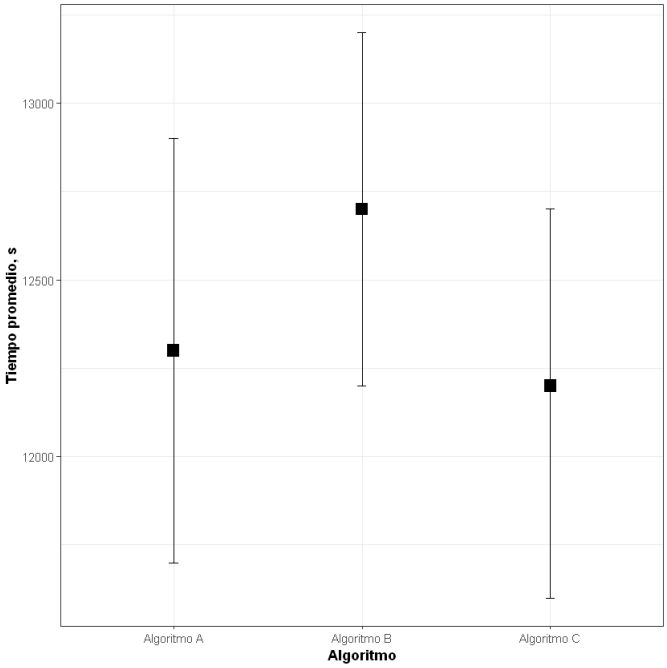
Algoritmo	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	40	12300	0.95	11700	12900
Algoritmo B	40	12700	0.95	12200	13200
Algoritmo C	40	12200	0.95	11600	12700

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```



A data.frame: 6 × 7

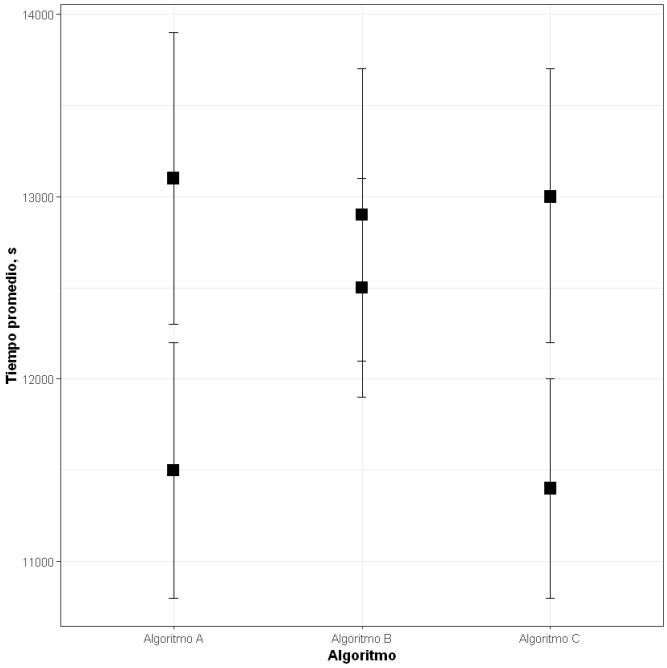
Algoritmo	Computadora	n	Mean	Conf.level	Percentile.lower	Percentile.upper
<fct>	<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	Computadora 1	20	13100	0.95	12300	13900
Algoritmo A	Computadora 2	20	11500	0.95	10800	12200
Algoritmo B	Computadora 1	20	12500	0.95	11900	13100
Algoritmo B	Computadora 2	20	12900	0.95	12100	13700
Algoritmo C	Computadora 1	20	11400	0.95	10800	12000
Algoritmo C	Computadora 2	20	13000	0.95	12200	13700

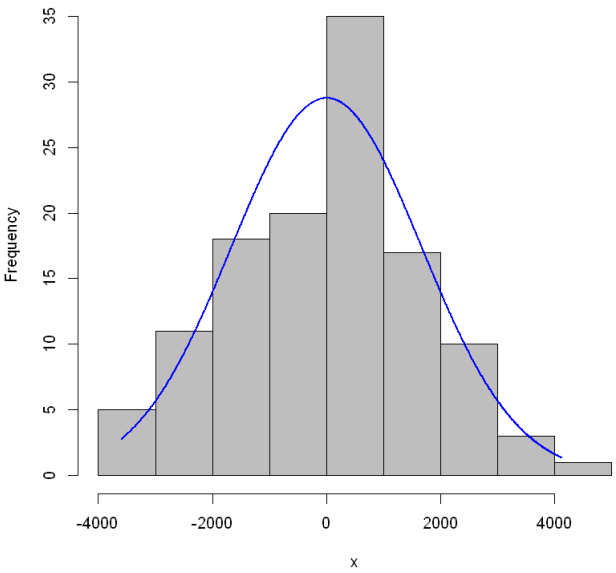


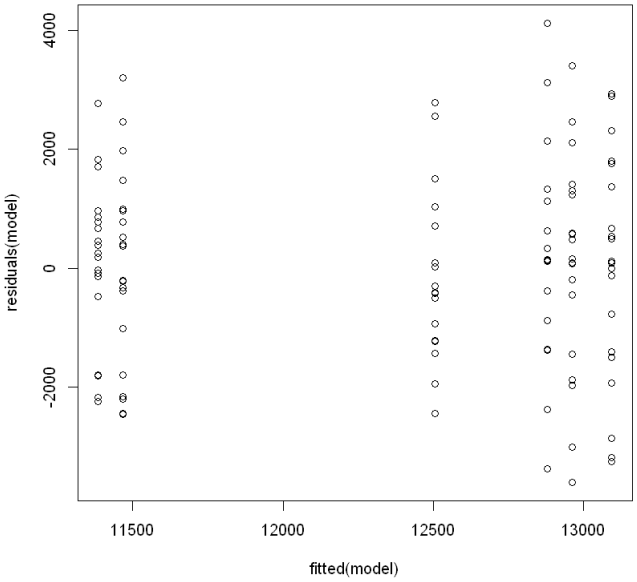

```
Call:
lm(formula = Tiempo ~ Algoritmo:Computadora, data = Data)

Residuals:
    Min      1Q  Median      3Q     Max
-3597.7 -1364.9   106.3  1038.4  4121.1

Coefficients: (1 not defined because of singularities)
                                          Estimate Std. Error t value
(Intercept)                                12961.7      380.4  34.074
AlgoritmoAlgoritmo A:ComputadoraComputadora 1    131.9      538.0   0.245
AlgoritmoAlgoritmo B:ComputadoraComputadora 1   -456.1      538.0  -0.848
AlgoritmoAlgoritmo C:ComputadoraComputadora 1  -1576.1      538.0  -2.930
AlgoritmoAlgoritmo A:ComputadoraComputadora 2  -1494.7      538.0  -2.778
AlgoritmoAlgoritmo B:ComputadoraComputadora 2    -82.8      538.0  -0.154
AlgoritmoAlgoritmo C:ComputadoraComputadora 2       NA         NA      NA
                                          Pr(>|t|)
(Intercept)                                < 2e-16 ***
AlgoritmoAlgoritmo A:ComputadoraComputadora 1  0.80668
AlgoritmoAlgoritmo B:ComputadoraComputadora 1  0.39832
AlgoritmoAlgoritmo C:ComputadoraComputadora 1  0.00410 **
AlgoritmoAlgoritmo A:ComputadoraComputadora 2  0.00639 **
AlgoritmoAlgoritmo B:ComputadoraComputadora 2  0.87795
AlgoritmoAlgoritmo C:ComputadoraComputadora 2       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1701 on 114 degrees of freedom
Multiple R-squared:  0.151,     Adjusted R-squared:  0.1138
F-statistic: 4.056 on 5 and 114 DF,  p-value: 0.002001

Note: model has aliased coefficients
      sums of squares computed by model comparison
```

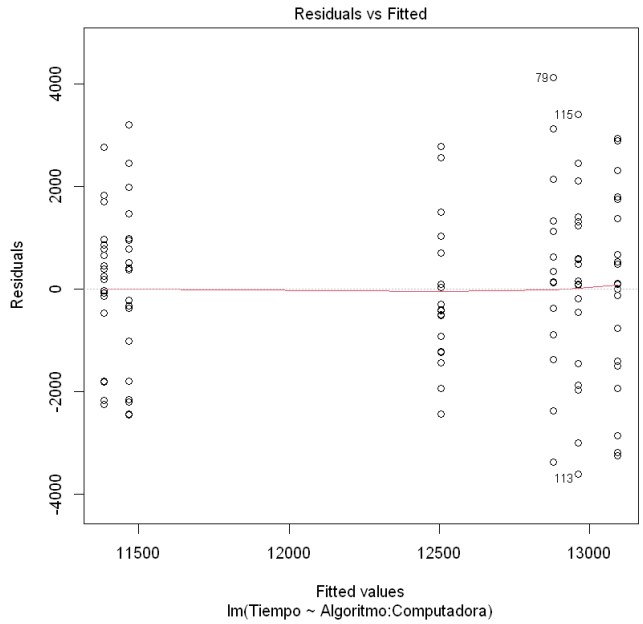
A anova: 2 × 4

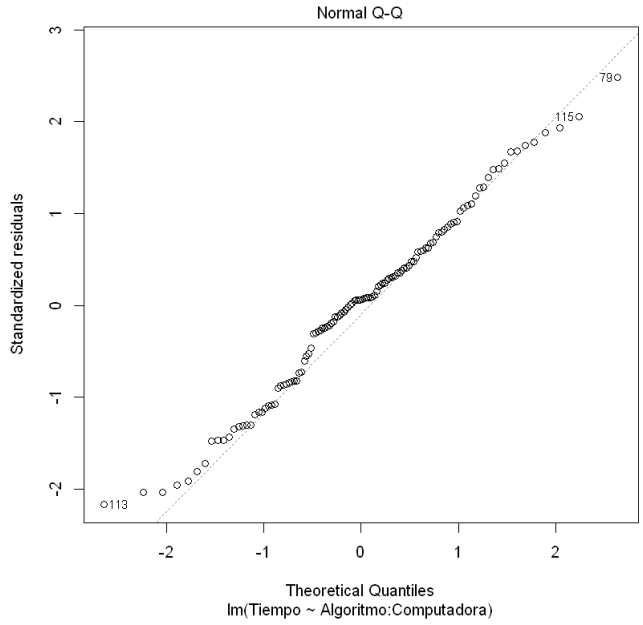
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo:Computadora	58692616	5	4.056014	0.002001063
Residuals	329927781	114	NA	NA

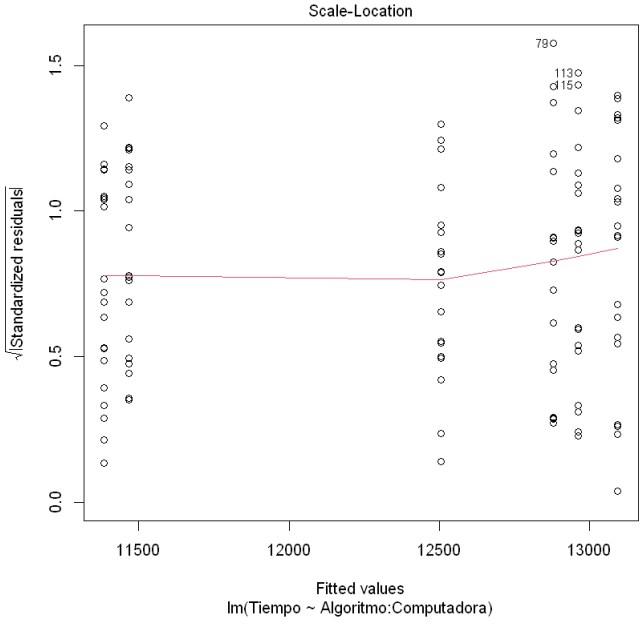

14000
13000
12000
11000
Tiempo promedio, s
Algoritmo A
Algoritmo B
Algoritmo C
Algoritmo


Frequency
0
5
10
15
20
25
30
35
-4000
-2000
0
2000
4000
x


residuals(model)
-2000
0
2000
4000
11500
12000
12500
13000
fitted(model)


Residuals vs Fitted
Residuals
Fitted values
lm(Tiempo ~ Algoritmo:Computadora)
79
115
113
4000
2000
0
-2000
-4000
11500
12000
12500
13000


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Tiempo ~ Algoritmo:Computadora)
79
115
113
3
2
1
0
-1
-2


Scale-Location
√|Standardized residuals|
Fitted values
lm(Tiempo ~ Algoritmo:Computadora)
79
113
115
1.5
1.0
0.5
0.0
11500
12000
12500
13000

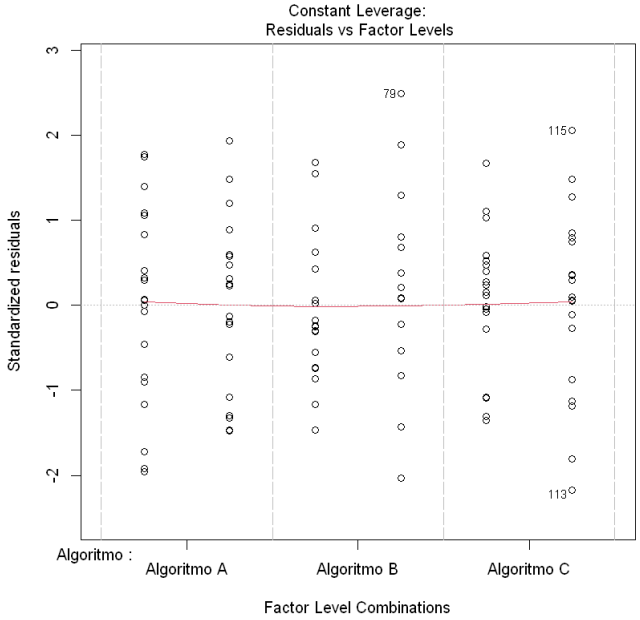
```
 contrast                                                estimate  SE  df t.ratio
 Algoritmo A Computadora 1 - Algoritmo B Computadora 1    588.0 538 114   1.093
 Algoritmo A Computadora 1 - Algoritmo C Computadora 1   1708.0 538 114   3.175
 Algoritmo A Computadora 1 - Algoritmo A Computadora 2   1626.6 538 114   3.024
 Algoritmo A Computadora 1 - Algoritmo B Computadora 2    214.8 538 114   0.399
 Algoritmo A Computadora 1 - Algoritmo C Computadora 2    131.9 538 114   0.245
 Algoritmo B Computadora 1 - Algoritmo C Computadora 1   1120.0 538 114   2.082
 Algoritmo B Computadora 1 - Algoritmo A Computadora 2   1038.5 538 114   1.931
 Algoritmo B Computadora 1 - Algoritmo B Computadora 2   -373.3 538 114  -0.694
 Algoritmo B Computadora 1 - Algoritmo C Computadora 2   -456.1 538 114  -0.848
 Algoritmo C Computadora 1 - Algoritmo A Computadora 2    -81.5 538 114  -0.151
 Algoritmo C Computadora 1 - Algoritmo B Computadora 2  -1493.3 538 114  -2.776
 Algoritmo C Computadora 1 - Algoritmo C Computadora 2  -1576.1 538 114  -2.930
 Algoritmo A Computadora 2 - Algoritmo B Computadora 2  -1411.8 538 114  -2.624
 Algoritmo A Computadora 2 - Algoritmo C Computadora 2  -1494.7 538 114  -2.778
 Algoritmo B Computadora 2 - Algoritmo C Computadora 2    -82.8 538 114  -0.154
 p.value
  0.8832
  0.0231
  0.0356
  0.9987
  0.9999
  0.3041
  0.3892
  0.9823
  0.9577
  1.0000
  0.0688
  0.0460
  0.0997
  0.0684
  1.0000

P value adjustment: tukey method for comparing a family of 6 estimates
```

```
Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 6 × 8

	Algoritmo	Computadora	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	Algoritmo C	Computadora 1	11385.60	380.4013	114	10367.07	12404.13	a
4	Algoritmo A	Computadora 2	11467.05	380.4013	114	10448.52	12485.58	ab
2	Algoritmo B	Computadora 1	12505.60	380.4013	114	11487.07	13524.13	abc
5	Algoritmo B	Computadora 2	12878.90	380.4013	114	11860.37	13897.43	abc
6	Algoritmo C	Computadora 2	12961.70	380.4013	114	11943.17	13980.23	bc
1	Algoritmo A	Computadora 1	13093.65	380.4013	114	12075.12	14112.18	c

png: 2


Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
3
2
1
0
-1
-2
79
115
113
Algoritmo :
Algoritmo A
Algoritmo B
Algoritmo C
Factor Level Combinations

In [ ]:

In [1]:

```r
# Kruskal-Wallis no paramétrica.

# Librerías.
if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(lattice)){install.packages("lattice")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(rcompanion)){install.packages("rcompanion")}


# 1. Carga inicial de datos.
ln <- ("
Algoritmo         Rendimiento
'Algoritmo A'         3
'Algoritmo A'         5
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         5
'Algoritmo A'         5
'Algoritmo A'         3
'Algoritmo A'         5
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         4
'Algoritmo A'         5
'Algoritmo A'         5
'Algoritmo B'         2
'Algoritmo B'         4
'Algoritmo B'         2
'Algoritmo B'         2
'Algoritmo B'         1
'Algoritmo B'         2
'Algoritmo B'         3
'Algoritmo B'         2
'Algoritmo B'         2
'Algoritmo B'         3
'Algoritmo B'         2
'Algoritmo B'         4
'Algoritmo B'         2
'Algoritmo B'         2
'Algoritmo B'         1
'Algoritmo B'         2
'Algoritmo B'         3
'Algoritmo B'         2
'Algoritmo B'         2
'Algoritmo B'         3
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         5
'Algoritmo C'         3
'Algoritmo C'         5
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         3
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         5
'Algoritmo C'         3
'Algoritmo C'         5
'Algoritmo C'         4
'Algoritmo C'         4
'Algoritmo C'         3")
```

```r
# Se introduce la tabla.
Data <- read.table(textConnection(ln), header=TRUE)
# Ordenamos los factores y craemos una variable puntaje del rendimiento.
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))
Data$Rendimiento.f <- factor(Data$Rendimiento, ordered = TRUE) # Variable puntaje.

# 2. Verificación de la lectura de datos.
library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(ln)

# 3. Resumimos la tabla.
xtabs(~ Algoritmo + Rendimiento.f, data = Data)

# Ponderación entre 0 a 1.
XT <- xtabs(~ Algoritmo + Rendimiento.f, data = Data)
prop.table(XT, margin = 1)

# 4. Gráfico de barras por grupo.
library(lattice)
histogram(~ Rendimiento.f | Algoritmo, data = Data, layout = c(1,3))

# 5. Resumen.
library(FSA)
Summarize(Rendimiento ~ Algoritmo, data =  Data, digits = 3)

# 6. Prueba Kruskal-Wallis.
kruskal.test(Rendimiento ~ Algoritmo, data = Data)

# 7. Análisis post-hoc.
Data$Algoritmo <- factor(Data$Algoritmo, levels = c("Algoritmo A",
                                                    "Algoritmo B",
                                                    "Algoritmo C"))
levels(Data$Algoritmo)

library(FSA)
DT <- dunnTest(Rendimiento ~ Algoritmo, data = Data, method = "bh")
DT

# 8. Despliege compacto con letras.
PT <- DT$res
PT
library(rcompanion)
cldList(P.adj ~ Comparison, data = PT, threshold = 0.05)

# 9. Gráfico de medianas en intervalos de confianza.
library(rcompanion)
Sum <- groupwiseMedian(Rendimiento ~ Algoritmo,
                       data = Data, conf = 0.95, R = 5000, percentile = TRUE,
                       bca = FALSE, digits = 3)
Sum

X <- 1:3
Y <- Sum$Percentile.upper + 0.2
Label <- c("a", "b", "a")
library(ggplot2)
ggplot(Sum, aes(x = Algoritmo, y = Median)) + geom_errorbar(aes(ymin = Percentile.lower,
                                                                ymax = Percentile.upper),
       width = 0.05, size = 0.5) +
       geom_point(shape = 15, size = 4) +
       theme_bw() +
       theme(axis.title = element_text(face = "bold")) +
       ylab("Mediana de puntaje de rendimiento") +
       annotate("text", x = X, y = Y, label = Label)
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: lattice

Loading required package: multcompView

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 3

	Algoritmo	Rendimiento	Rendimiento.f
	<fct>	<chr>	<ord>
1	Algoritmo A	3	3
2	Algoritmo A	5	5
3	Algoritmo A	4	4
4	Algoritmo A	4	4
...	NA	...	NA
57	Algoritmo C	5	5
58	Algoritmo C	4	4
59	Algoritmo C	4	4
60	Algoritmo C	3	3

```
'data.frame':   60 obs. of  3 variables:
 $ Algoritmo    : Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Rendimiento  : int  3 5 4 4 4 4 4 4 5 5 ...
 $ Rendimiento.f: Ord.factor w/ 5 levels "1"<"2"<"3"<"4"<..: 3 5 4 4 4 4 4 4 5 5 ...
      Algoritmo   Rendimiento  Rendimiento.f
 Algoritmo A:20   Min.   :1.0   1: 2
 Algoritmo B:20   1st Qu.:3.0   2:12
 Algoritmo C:20   Median :4.0   3:10
                  Mean   :3.5   4:26
                  3rd Qu.:4.0   5:10
                  Max.   :5.0
             Rendimiento.f
Algoritmo      1  2  3  4  5
  Algoritmo A  0  0  2 12  6
  Algoritmo B  2 12  4  2  0
  Algoritmo C  0  0  4 12  4
             Rendimiento.f
Algoritmo       1   2   3   4   5
  Algoritmo A 0.0 0.0 0.1 0.6 0.3
  Algoritmo B 0.1 0.6 0.2 0.1 0.0
  Algoritmo C 0.0 0.0 0.2 0.6 0.2
```

A data.frame: 3 × 9

Algoritmo	n	mean	sd	min	Q1	median	Q3	max
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	4.2	0.616	3	4	4	5	5
Algoritmo B	20	2.3	0.801	1	2	2	3	4
Algoritmo C	20	4.0	0.649	3	4	4	4	5

```
        Kruskal-Wallis rank sum test

data:  Rendimiento by Algoritmo
Kruskal-Wallis chi-squared = 34.265, df = 2, p-value = 3.625e-08
```

'Algoritmo A' · 'Algoritmo B' · 'Algoritmo C'

```
Dunn (1964) Kruskal-Wallis multiple comparison

  p-values adjusted with the Benjamini-Hochberg method.
```

```
                 Comparison          Z      P.unadj        P.adj
1 Algoritmo A - Algoritmo B  5.3776947 7.544560e-08 2.263368e-07
2 Algoritmo A - Algoritmo C  0.6865142 4.923889e-01 4.923889e-01
3 Algoritmo B - Algoritmo C -4.6911805 2.716332e-06 4.074498e-06
```

A data.frame: 3 × 4

Comparison	Z	P.unadj	P.adj
<chr>	<dbl>	<dbl>	<dbl>
Algoritmo A - Algoritmo B	5.3776947	7.544560e-08	2.263368e-07
Algoritmo A - Algoritmo C	0.6865142	4.923889e-01	4.923889e-01
Algoritmo B - Algoritmo C	-4.6911805	2.716332e-06	4.074498e-06

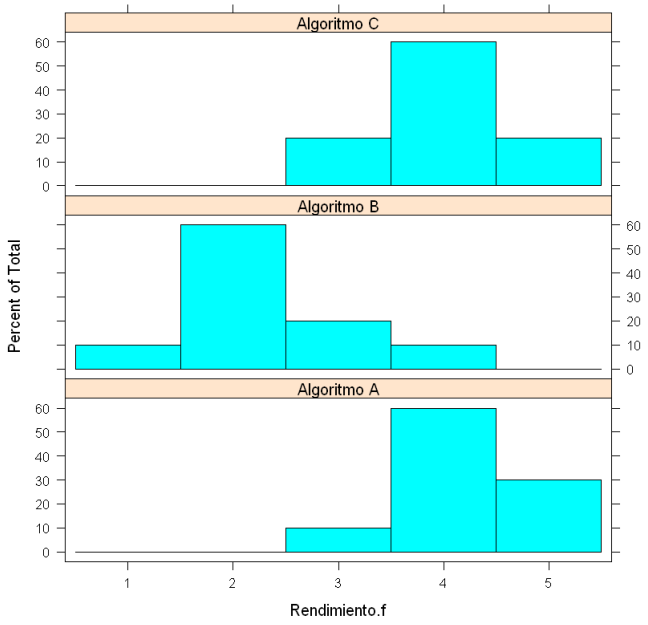
A data.frame: 3 × 3

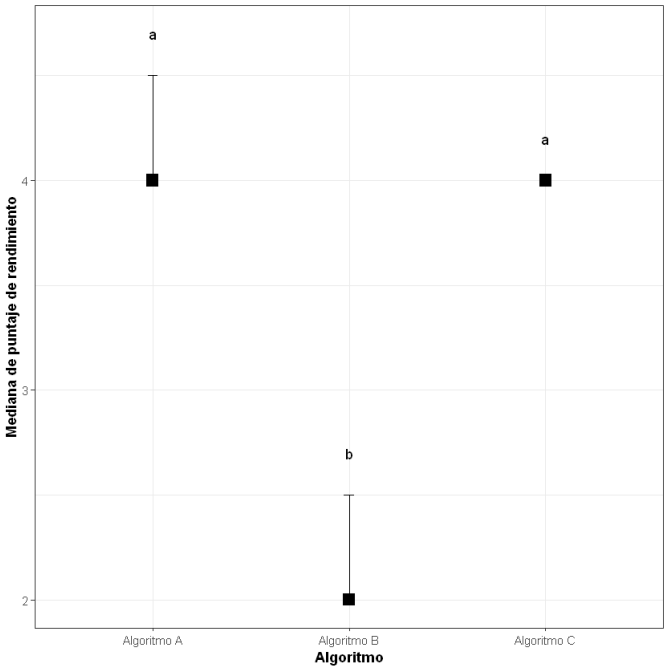
Group	Letter	MonoLetter
<chr>	<chr>	<chr>
AlgoritmoA	a	a
AlgoritmoB	b	b
AlgoritmoC	a	a

A data.frame: 3 × 6

Algoritmo	n	Median	Conf.level	Percentile.lower	Percentile.upper
<fct>	<int>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	4	0.95	4	4.5
Algoritmo B	20	2	0.95	2	2.5
Algoritmo C	20	4	0.95	4	4.0

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```





In [ ]:

In [ ]:

```r
# Bibliotecas
if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}

# 1. Carga de datos.
Data <- read.csv2("Datos_fixed_tarea_2_fix.csv", sep = ";", header = TRUE)
Data$Efectos <- as.factor(Data$Efectos)
Data$Objetos <- as.character(Data$Objetos)
Data$Objectos <- as.factor(Data$Objetos)
Data$Arquitectura <- as.factor(Data$Arquitectura)
Data$Resolucion <- as.factor(Data$Resolucion)
Data$Tiempo <- as.numeric(Data$Tiempo)

# 2. Verificación de la lectura de datos.
# Verificar que solo devuelva métricas para el tiempo.
# Si sale NA, factor no definido como tal.
library(psych)
headTail(Data)
str(Data)
summary(Data)

# 3. Gráficos simples de interacción
# Variable dependiente: Tiempo
# Variables independientes: Arquitectura y Objetos
interaction.plot(x.factor = as.numeric(Data$Objetos),
 trace.factor = Data$Arquitectura,
 response = Data$Tiempo,
 fun = mean,
 type = "b",
 col = c("black", "red", "green"),
 pch = c(19,17,15),
 fixed = TRUE,
 leg.bty = "o")
# Parece haber comportamiento exponencial.
# Pero el eje X dice cuantos objetos tenía la escena.
# Son potencias de 2.
# Se debe analizar si estamos forzando un comportamiento en alguno de los factores.
# APU se comporta mejor.

# Variable dependiente: Tiempo
# Variables independientes: Arquitectura y Resolucion
interaction.plot(x.factor = Data$Resolucion,
 trace.factor = Data$Arquitectura,
 response = Data$Tiempo,
 fun = mean,
 type = "b",
 col = c("black", "red", "green"),
 pch = c(19,17,15),
 fixed = TRUE,
 leg.bty = "o")

# Variable dependiente: Tiempo
# Variables independientes: Arquitectura y Efectos
interaction.plot(x.factor = Data$Efectos,
 trace.factor = Data$Arquitectura,
 response = Data$Tiempo,
 fun = mean,
 type = "b",
 col = c("black", "red", "green"),
 pch = c(19,17,15),
 fixed = TRUE,
 leg.bty = "o")
# Si hay interacción significativa entre objetos y efectos, los objetos
# impactan en cómo se comportan los efectos.

# 4. Evaluación de los supuestos.
# Función para graficos de los supuestos.
graficos_supuestos <- function(model) {
  par(mfrow = c(3, 1))
  x <- residuals(model)
```

```r
  library(rcompanion)
  plotNormalHistogram(x)
  plot(fitted(model), residuals(model))
  qqnorm(resid(model), main = "Normal Q-Q", xlab = "Theoretical Quantiles", ylab = "Standarized residuals")
  qqline(resid(model), col = "red", lwd = 2)
  par(mfrow = c(1, 1))
}

# Datos iniciales originales.
model <- lm(Tiempo ~ Objetos * Arquitectura * Efectos * Resolucion, data = Data)
graficos_supuestos(model)
leveneTest(Tiempo ~ Objetos * Arquitectura * Efectos * Resolucion, data = Data)

# 5. Tranformación por raíz cuadrada.
library(rcompanion)
T_sqrt <- sqrt(Data$Tiempo)
model <- lm(T_sqrt ~ Objetos * Arquitectura * Efectos * Resolucion, data = Data)
graficos_supuestos(model)
leveneTest(T_sqrt ~ Objetos * Arquitectura * Efectos * Resolucion, data = Data)
plot(model)

# 6. Anova
library(car)
Anova(model, type = "II")

# 7. Gráficos finales
# Arquitectura
Sum <- Summarize(T_sqrt ~ Arquitectura, data = Data, digits = 3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Arquitectura, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0),
            legend.position="none") +
      ylab(expression("Promerdio de la raíz cuadrada del tiempo")) +
      ggtitle("Tiempo vs Arquitectura")

# Destransformando
Sum <- Summarize(T_sqrt ~ Arquitectura, data = Data, digits = 3)
Sum$mean <- Sum$mean^2
Sum$sd <- Sum$sd^2
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Arquitectura, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0),
            legend.position="none") +
      ylab(expression("Tiempo promedio (s)")) +
      ggtitle("Tiempo vs Arquitectura")

# Arquitectura + Resolucion
Sum <- Summarize(T_sqrt ~ Arquitectura + Resolucion, data = Data, digits = 3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)

library(ggplot2)
```

```r
pd <- position_dodge(.2)
 ggplot(Sum,aes(x=Resolucion, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Promerdio de la raíz cuadrada del tiempo")) +
      ggtitle("Tiempo vs Arquitectura")

# Destransformando
Sum <- Summarize(T_sqrt ~ Arquitectura + Resolucion, data = Data, digits = 3)
Sum$mean <- Sum$mean^2
Sum$sd <- Sum$sd^2
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)

library(ggplot2)
pd <- position_dodge(.2)
 ggplot(Sum,aes(x=Resolucion, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Tiempo promedio (s)")) +
      ggtitle("Tiempo vs Arquitectura")

# Arquitectura y efectos
Sum <- Summarize(T_sqrt ~ Arquitectura + Efectos, data = Data, digits = 3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum$Efectos <- factor(Sum$Efectos,
                      levels(Sum$Efectos)[c(8,7,6,5,4,3,2,1)])

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Efectos, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Promerdio de la raíz cuadrada del tiempo")) +
      ggtitle("Tiempo vs Arquitectura")

# Para salvar
# ggsave(plot = q, width = 14, height = 8, dpi = 300, filename = "arquitectura.png")
# Destransformando
Sum <- Summarize(T_sqrt ~ Arquitectura + Efectos, data = Data, digits = 3)
Sum$mean <- Sum$mean^2
Sum$sd <- Sum$sd^2
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum$Efectos <- factor(Sum$Efectos,
                      levels(Sum$Efectos)[c(8,7,6,5,4,3,2,1)])

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Efectos, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
```

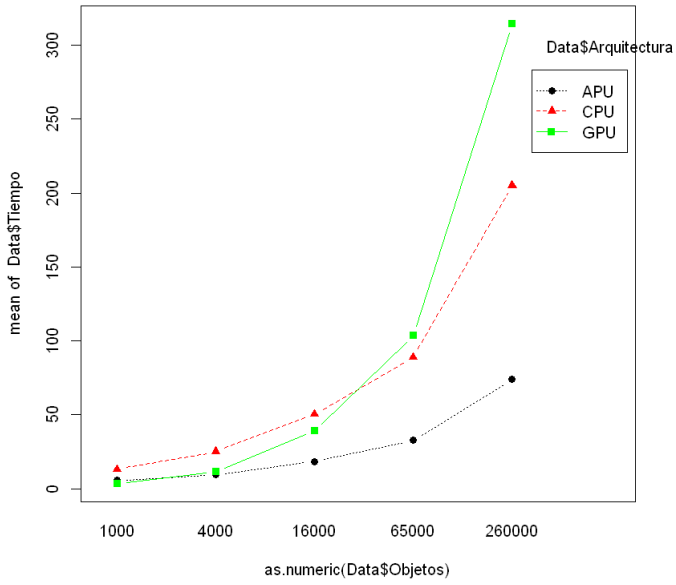
```
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Tiempo promedio (s)")) +
      ggtitle("Tiempo vs Arquitectura")

# Arquitectura y Objetos
Sum <- Summarize(T_sqrt ~ Arquitectura + Objetos, data = Data, digits = 3)
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)

library(ggplot2)
pd <- position_dodge(.2)
ggplot(Sum,aes(x=Objetos, y=mean, color = Arquitectura)) + geom_errorbar(aes(ymin =
      mean - se,ymax = mean + se),width=.2,size=0.7, position=pd)+
      geom_point(aes(shape=Arquitectura), size=5, position=pd)+ theme_bw() +
      theme(plot.title = element_text(face="bold", hjust=0.5),
            axis.title = element_text(face="bold"),
            axis.text = element_text(face="bold"),
            plot.caption= element_text(hjust=0),
            legend.text = element_text(face="bold"),
            legend.title = element_text(face="bold"),
            legend.justification = c(1,0)) +
      ylab(expression("Promerdio de la raíz cuadrada del tiempo")) +
      ggtitle("Tiempo vs Arquitectura")

# Destransformando es la misma vara

# 8. Pairwise t-test
pairwise.t.test(T_sqrt, Data$Arquitectura, p.adjust.method = "BH")
pairwise.t.test(T_sqrt, Data$Arquitectura : Data$Resolucion, p.adjust.method = "BH")
pairwise.t.test(T_sqrt, Data$Arquitectura : Data$Efectos, p.adjust.method = "BH")
# Se pueden hacer análisis de todas las interacciones que se quieran.

# 9. Conclusión.
# 1. En la totalidad de experimentos, el APU se comportó mejor.
# 2. Se identificó que para escenarios con pocos objetos, no hay diferencia. En escenarios donde la cantidad
# aumenta significativamente, entre más objetos hallan mejor el APU con respecto a las otras dos. En escenar
# no hay diferencia, pero en escenarios complejos si.
```

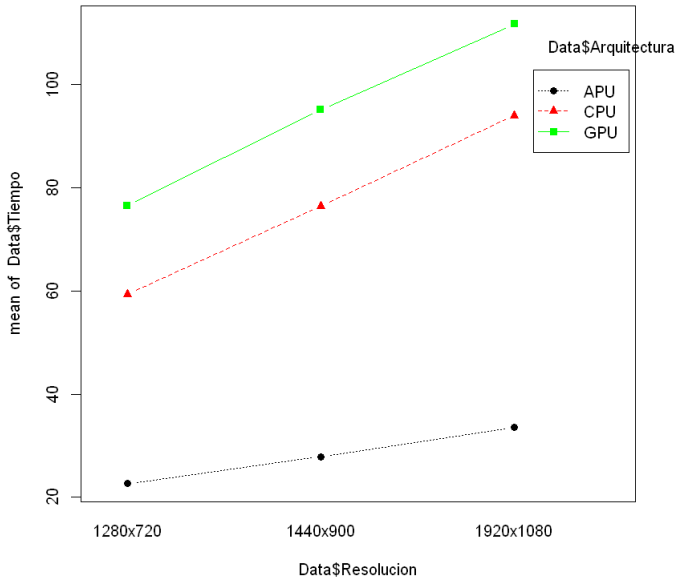
```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

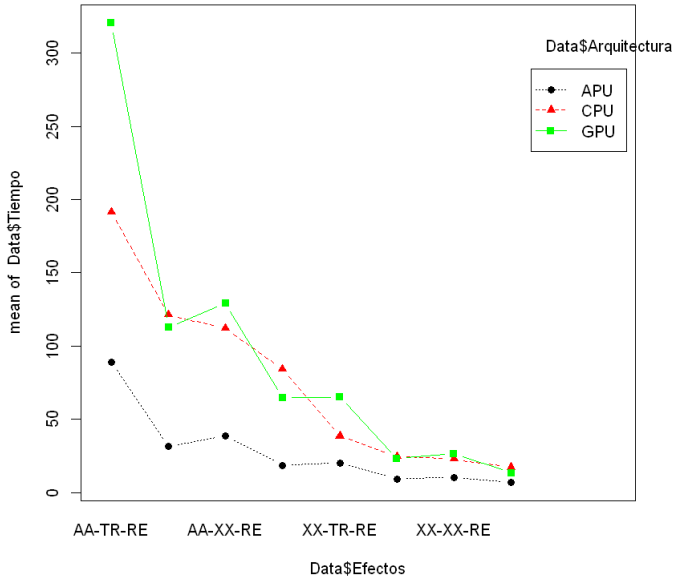
A data.frame: 9 × 6

	Tiempo	Objetos	Arquitectura	Efectos	Resolucion	Objectos
	<chr>	<chr>	<fct>	<fct>	<fct>	<fct>
1	6.53	16000	APU	XX-TR-XX	1280x720	16000
2	4.22	1000	APU	XX-TR-RE	1440x900	1000
3	6.13	1000	APU	AA-XX-RE	1280x720	1000
4	21.75	16000	APU	AA-TR-XX	1440x900	16000
...	...	NA	NA	NA	NA	NA
1797	11.75	16000	GPU	XX-TR-XX	1920x1080	16000
1798	31.93	4000	GPU	AA-TR-RE	1440x900	4000
1799	47.3	260000	GPU	XX-XX-XX	1440x900	260000
1800	5.44	1000	GPU	AA-XX-RE	1440x900	1000

```
'data.frame':   1800 obs. of  6 variables:
 $ Tiempo      : num  6.53 4.22 6.13 21.75 4.47 ...
 $ Objetos     : chr  "16000" "1000" "1000" "16000" ...
 $ Arquitectura: Factor w/ 3 levels "APU","CPU","GPU": 1 1 1 1 1 1 1 1 1 1 ...
 $ Efectos     : Factor w/ 8 levels "AA-TR-RE","AA-TR-XX",..: 6 5 3 2 7 7 2 8 1 2 ...
 $ Resolucion  : Factor w/ 3 levels "1280x720","1440x900",..: 1 2 1 2 1 1 2 1 2 3 ...
 $ Objectos    : Factor w/ 5 levels "1000","16000",..: 2 1 1 2 4 3 3 5 1 4 ...
    Tiempo           Objetos          Arquitectura     Efectos     
 Min.   :   0.880   Length:1800        APU:600      AA-TR-RE:225  
 1st Qu.:   7.315   Class :character   CPU:600      AA-TR-XX:225  
 Median :  20.130   Mode  :character   GPU:600      AA-XX-RE:225  
 Mean   :  66.326                                   AA-XX-XX:225  
 3rd Qu.:  64.073                                   XX-TR-RE:225  
 Max.   :1271.990                                   XX-TR-XX:225  
                                                    (Other) :450  

    Resolucion    Objectos   
 1280x720 :600   1000  :360  
 1440x900 :600   16000 :360  
 1920x1080:600   260000:360  
                 4000  :360  
                 65000 :360  
```

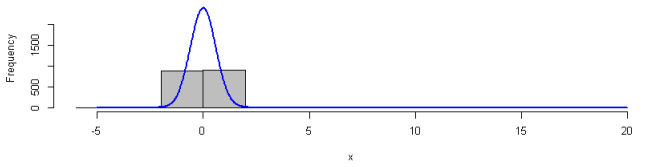


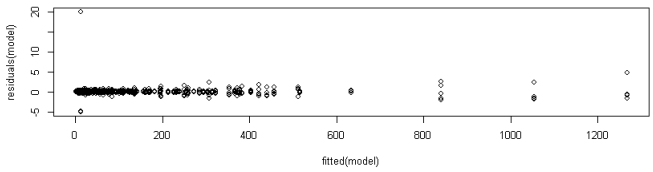


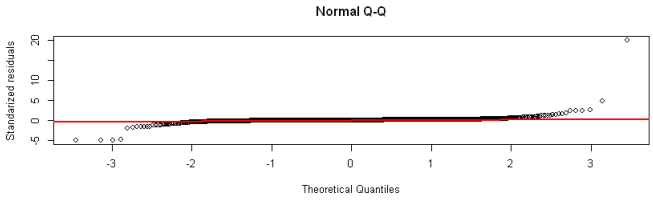


A anova: 2 × 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	359	1.298032	0.0006341896
	1440	NA	NA

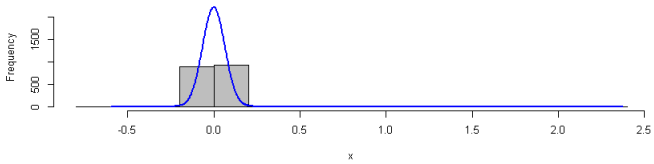


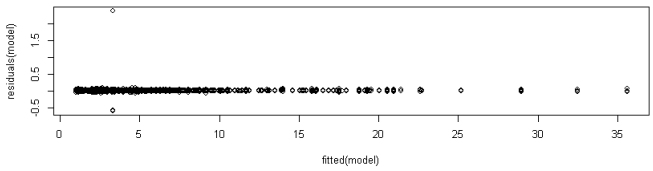


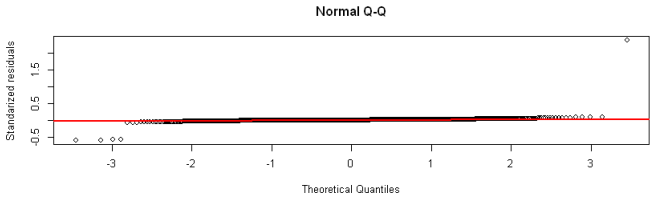


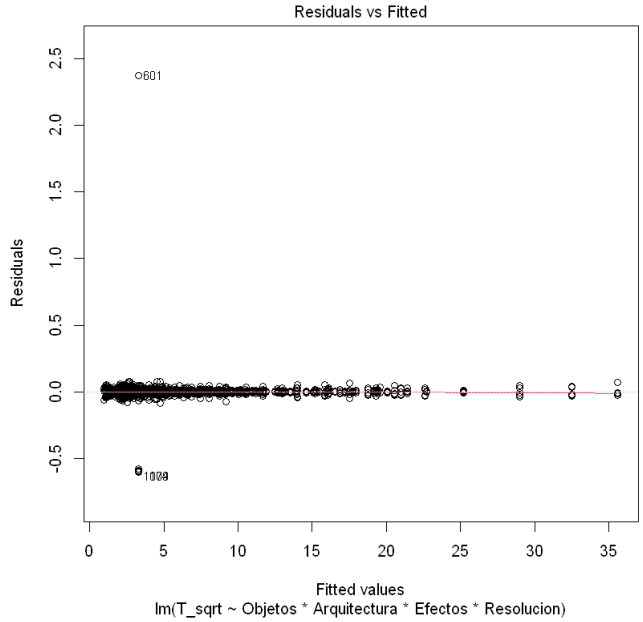
A anova: 2 × 3

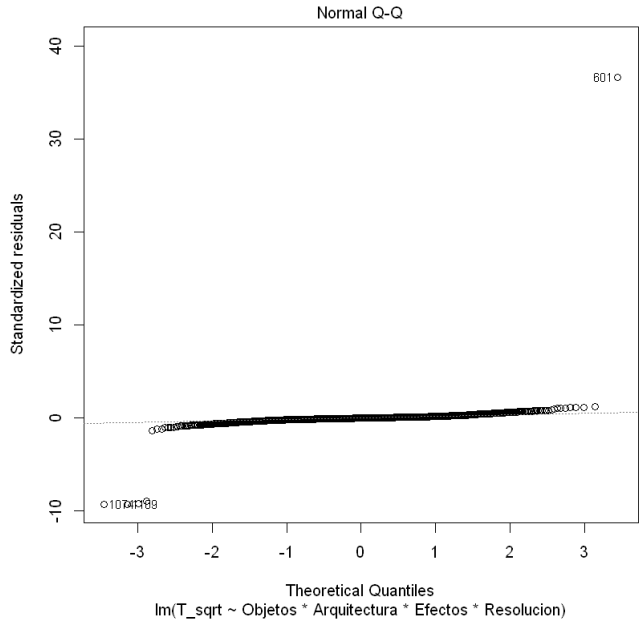
	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	359	1.026171	0.3718046
	1440	NA	NA

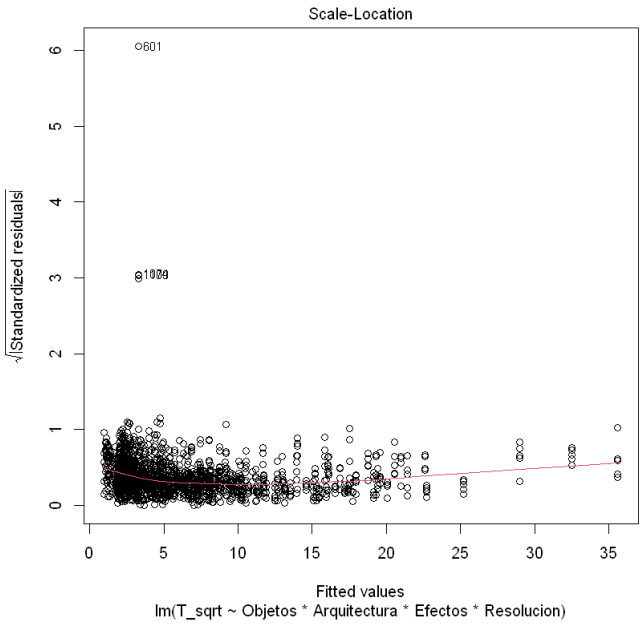







Residuals vs Fitted
Residuals
Fitted values
lm(T_sqrt ~ Objetos * Arquitectura * Efectos * Resolucion)

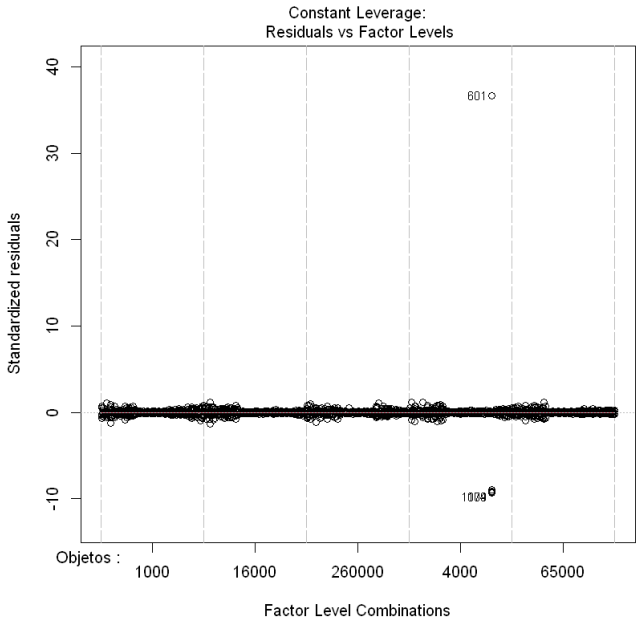

Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(T_sqrt ~ Objetos * Arquitectura * Efectos * Resolucion)

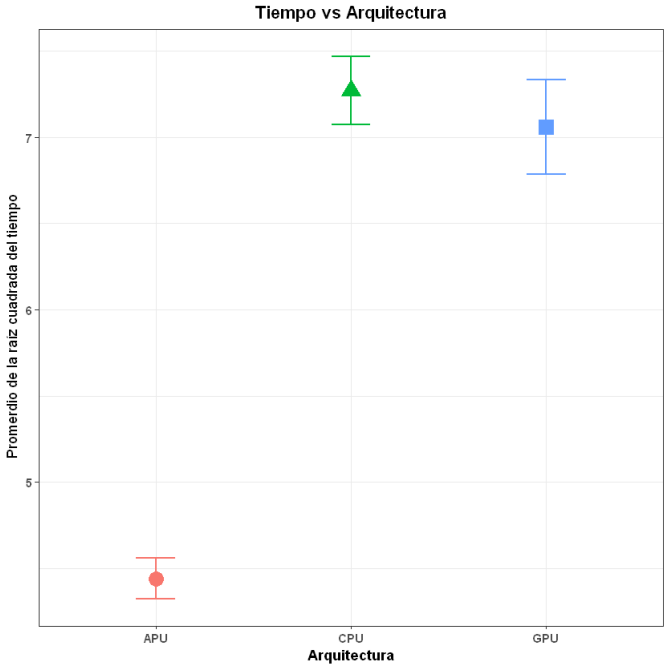

Scale-Location
√|Standardized residuals|
Fitted values
lm(T_sqrt ~ Objetos * Arquitectura * Efectos * Resolucion)

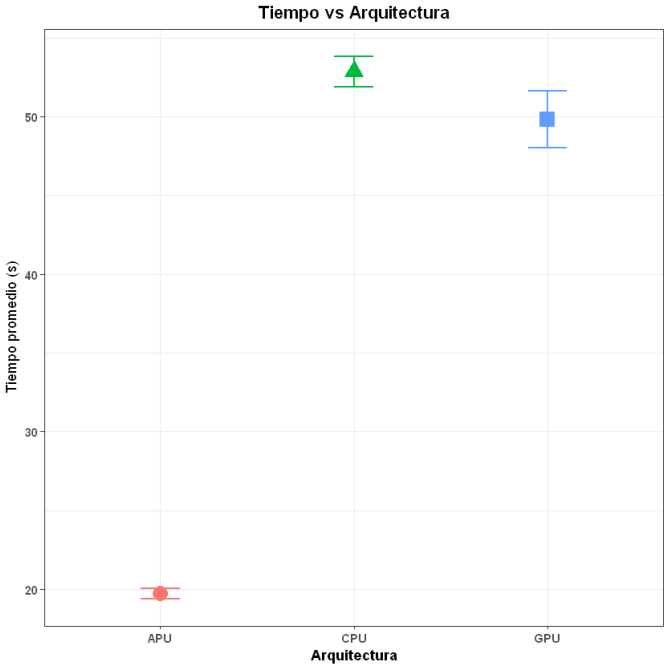
A anova: 16 × 4

	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Objetos	21603.355587	4	1.030566e+06	0.000000e+00
Arquitectura	2980.900957	2	2.844016e+05	0.000000e+00
Efectos	12945.315384	7	3.528816e+05	0.000000e+00
Resolucion	463.209430	2	4.419385e+04	0.000000e+00
Objetos:Arquitectura	2944.326900	8	7.022803e+04	0.000000e+00
Objetos:Efectos	5538.622898	28	3.774489e+04	0.000000e+00
Arquitectura:Efectos	1284.558830	14	1.750816e+04	0.000000e+00
Objetos:Resolucion	150.970723	8	3.600951e+03	0.000000e+00
Arquitectura:Resolucion	41.517836	4	1.980565e+03	0.000000e+00
Efectos:Resolucion	111.457600	14	1.519134e+03	0.000000e+00
Objetos:Arquitectura:Efectos	750.722134	56	2.558030e+03	0.000000e+00
Objetos:Arquitectura:Resolucion	13.169713	16	1.570619e+02	1.497901e-301
Objetos:Efectos:Resolucion	57.364654	56	1.954658e+02	0.000000e+00
Arquitectura:Efectos:Resolucion	11.826925	28	8.059874e+01	6.509456e-271
Objetos:Arquitectura:Efectos:Resolucion	7.353165	112	1.252768e+01	2.586956e-145
Residuals	7.546543	1440	NA	NA

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead."
```




Tiempo vs Arquitectura
Promerdio de la raiz cuadrada del tiempo
7
6
5
APU
CPU
GPU
Arquitectura


Tiempo vs Arquitectura
Tiempo promedio (s)
50
40
30
20
APU
CPU
GPU
Arquitectura

In [ ]:

In [ ]:

```r
# 1. Carga de datos

library(FSA)
library(psych)
library(knitr)
library(rcompanion)
library(ggplot2)

# Lectura de datos
Data <- read.csv("C:\\Users\\user\\PycharmProjects\\RCheatSheet\\Examen1\\Datos tarea 1.csv")

# Sumario
summ <- Summarize(Stpbnd.2400.2482..S21..1. ~ Lot, data=Data, digits = 2)

# Se calculan los rangos
ranges <- tapply(Data$Stpbnd.2400.2482..S21..1., Data$Lot, range)
summ$range <- ranges

# Para mostrar solo: tamaño de muestra, mínimo, máximo, rango, media, promedio y desviación estándar.
summ_organized <- summ[, c("Lot", "n", "min", "max", "range", "median", "mean", "sd")]
kable(summ_organized, align = "l", format = "markdown", row.names = FALSE, caption = "Summary")

# 2. Histogramas iniciales

Control <- Data$Stpbnd.2400.2482..S21..1.[Data$Lot == "Control"]
exp1 <- Data$Stpbnd.2400.2482..S21..1.[Data$Lot == "Exp 1"]
exp2 <- Data$Stpbnd.2400.2482..S21..1.[Data$Lot == "Exp 2"]
exp3 <- Data$Stpbnd.2400.2482..S21..1.[Data$Lot == "Exp 3"]
exp4 <- Data$Stpbnd.2400.2482..S21..1.[Data$Lot == "Exp 4"]
exp5 <- Data$Stpbnd.2400.2482..S21..1.[Data$Lot == "Exp 5"]

par(mfrow = c(3, 1))
plotNormalHistogram(Control, main="Control",  xlim = c(4, 56), lwd=0.5, xlab="dB")
plotNormalHistogram(exp1, main="Exp 1",  xlim = c(4, 56), lwd=0.5, xlab="dB")
plotNormalHistogram(exp2, main="Exp 2",  xlim = c(4, 56), lwd=0.5, xlab="dB")
par(mfrow = c(3, 1))
plotNormalHistogram(exp3, main="Exp 3",  xlim = c(4, 56), lwd=0.5, xlab="dB")
plotNormalHistogram(exp4, main="Exp 4",  xlim = c(4, 56), lwd=0.5, xlab="dB")
plotNormalHistogram(exp5, main="Exp 5",  xlim = c(4, 56), lwd=0.5, xlab="dB")

# 3. Eliminar outliers

# IQR
eliminate_outliers <- function(mydata) {
  quartiles <- quantile(mydata, probs=c(.25, .75), na.rm = FALSE)
  IQR <- IQR(mydata)
  Lower <- quartiles[1] - 1.5 * IQR
  Upper <- quartiles[2] + 1.5 * IQR
  clean_data <- subset(mydata, mydata > Lower & mydata < Upper)
  return (clean_data)
}

Control <- eliminate_outliers(Control)
exp1 <- eliminate_outliers(exp1)
exp2 <- eliminate_outliers(exp2)
exp3 <- eliminate_outliers(exp3)
exp4 <- eliminate_outliers(exp4)
exp5 <- eliminate_outliers(exp5)

par(mfrow = c(2, 1))
plotNormalHistogram(Control, main="Control",  xlim = c(24, 29), lwd=0.5, xlab="dB")
plotNormalHistogram(exp1, main="Exp 1",  xlim = c(24, 29), lwd=0.5, xlab="dB")
par(mfrow = c(2, 1))
plotNormalHistogram(exp2, main="Exp 2",  xlim = c(24, 29), lwd=0.5, xlab="dB")
plotNormalHistogram(exp3, main="Exp 3",  xlim = c(24, 29), lwd=0.5, xlab="dB")
par(mfrow = c(2, 1))
plotNormalHistogram(exp4, main="Exp 4",  xlim = c(24, 29), lwd=0.5, xlab="dB")
plotNormalHistogram(exp5, main="Exp 5",  xlim = c(24, 29), lwd=0.5, xlab="dB")

# 3. Histogramas de colores sobrepuestos

par(mfrow = c(1, 1))
p1 <- hist(Control)
p2 <- hist(exp1)
p3 <- hist(exp2)
```

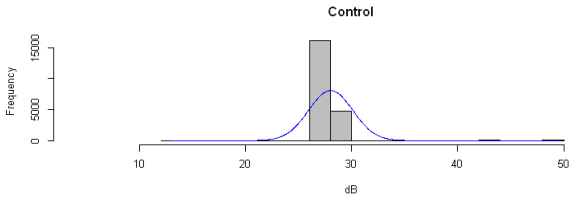
```r
plot( p1, col=rgb(0,0,1,1/4), xlim=c(24, 29), ylim=c(0, 8000), breaks = 30, xlab="dB",
     main = "Control - Exp1 - Exp2" )  # First histogram
plot( p2, col=rgb(1,0,0,1/4), breaks = 30, add=T) # Second histogram
plot( p3, col=rgb(0,1,0,1/4),  breaks = 30, add=T)  # Third histogram
legend("topright", c("Control", "Exp 1", "Exp 2"),
      fill = c(rgb(0,0,1,1/4),
               rgb(1,0,0,1/4),
               rgb(0,1,0,1/4)))

p4 <- hist(exp3)
p5 <- hist(exp4)
p6 <- hist(exp5)
plot( p1, col=rgb(0,0,1,1/4), xlim=c(24, 29), ylim=c(0, 8000), breaks = 30, xlab="dB",
       main = "Control - Exp3 - Exp4 - Exp5")  # first histogram
plot( p4, col=rgb(1, 0,1,1/4), breaks = 30, add=T)
plot( p5, col=rgb(1,1,0,1/4),  breaks = 30, add=T)
plot( p6, col=rgb(0,0,0,1/4),  breaks = 30, add=T)
legend("topright", c("Control", "Exp 3", "Exp 4", "Exp 5"),
       fill = c(rgb(0,0,1,1/4),
                rgb(1, 0,1,1/4),
                rgb(1,1,0,1/4),
                rgb(0,0,0,1/4)))


# 4. Gráfico de cajas y bigotes

# Para eliminar outliers "outline = 0".
boxplot(Stpbnd.2400.2482..S21..1. ~ Lot, data = Data, ylim = c(24, 30), ylab="Stopband (dB)")


# ---------------- PLAYGROUND -------------
# Promedios e intervalos de confianza

#  Stpbnd.2400.2482..S21..1. ~ Lot ||| data=Data,

Sum <- groupwiseMean(Stpbnd.2400.2482..S21..1. ~ Lot, data = Data, conf = 0.95, digits = 3, traditional = FA
Sum


# Gráficos de promedios e intervalos de confianza

library(ggplot2)
ggplot(Sum,
       aes(x = Lot, y = Mean)) +
       geom_errorbar(aes(ymin = Percentile.lower,
                         ymax = Percentile.upper),
                         width = 0.05, size = 0.5) +
                    geom_point(shape = 15,
                               size = 4) +
                    theme_bw() +
                    theme(axis.title = element_text(face = "bold")) +
                    ylab("Tiempo promedio, s")


model <- lm(Stpbnd.2400.2482..S21..1. ~ Lot, data = Data)
summary(model)


X <- residuals(model)
library(rcompanion)
plotNormalHistogram(X)

plot(fitted(model),residuals(model))

plot(model)


library(car)
Anova(model, type = "II")


library(multcompView)
library(lsmeans)
marginal <- lsmeans(model, ~ Lot)
pairs(marginal, adjust="tukey", alpha = 0.001)
```

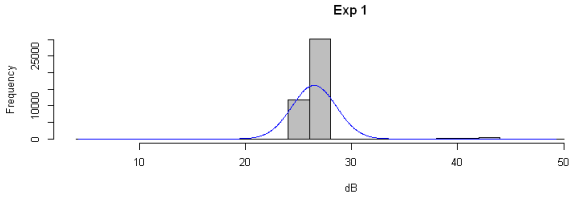
```
library(multcomp)
CLD <- cld(marginal, alpha=0.001, Letters = letters, adjust = "tukey")
CLD
```

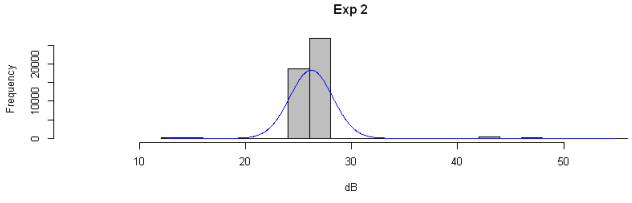
```
## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'psych'


The following object is masked from 'package:FSA':

    headtail


Warning message:
"package 'knitr' was built under R version 4.2.3"
Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi



Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha
```

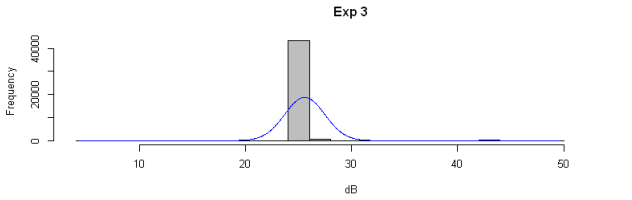
Table: Summary

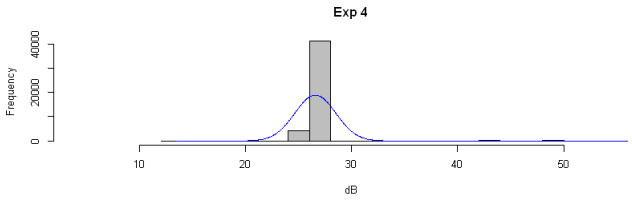
Lot	n	min	max	range	median	mean	sd
Control	21039	12.93	49.57	12.932, 49.570	27.81	28.02	2.08
Exp 1	42479	4.26	49.31	4.258, 49.307	26.24	26.46	2.09
Exp 2	46350	12.64	54.79	12.644, 54.794	26.06	26.19	2.02
Exp 3	44311	4.32	49.69	4.315, 49.690	25.40	25.57	1.88
Exp 4	45805	13.49	55.90	13.495, 55.897	26.46	26.57	1.93
Exp 5	44700	13.27	49.98	13.269, 49.982	25.33	25.52	1.95

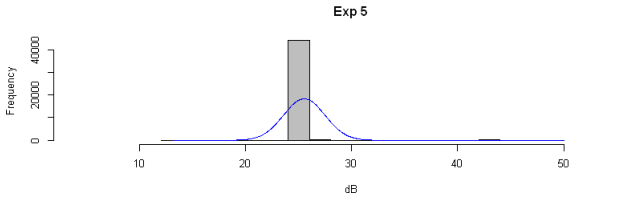


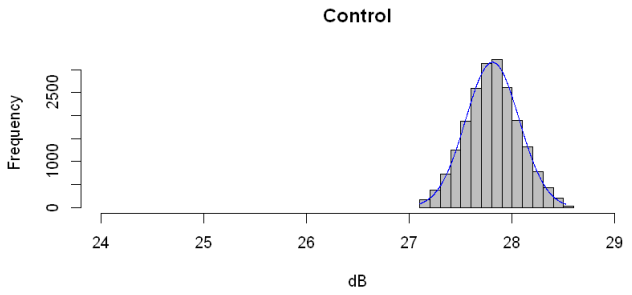


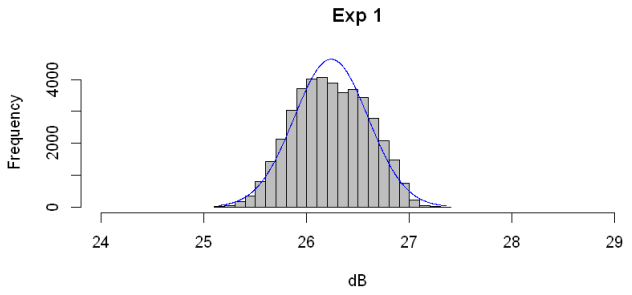


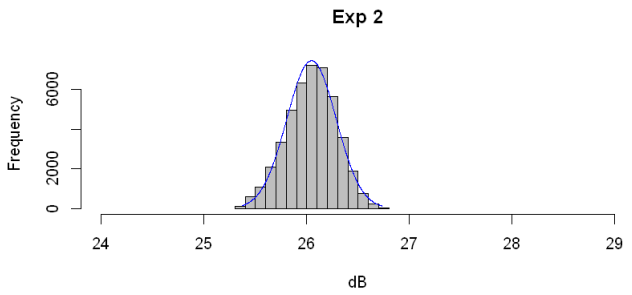

Exp 3
Frequency
0
20000
40000
10
20
30
40
50
dB

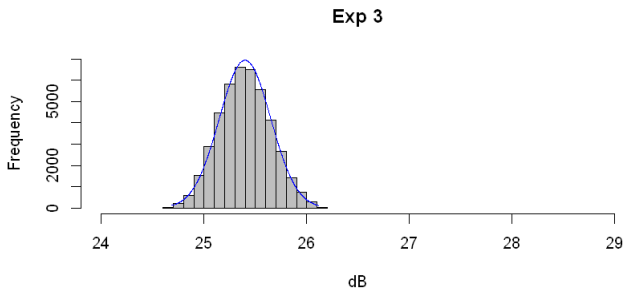

Exp 4
Frequency
0
20000
40000
10
20
30
40
50
dB

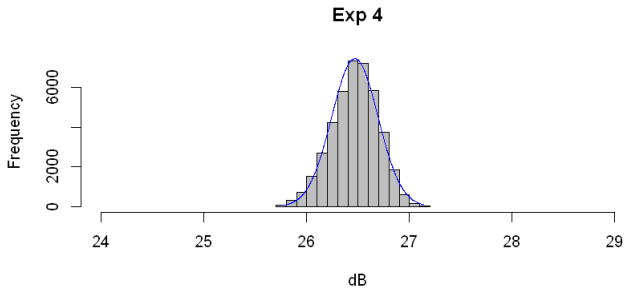

Exp 5
Frequency
0
20000
40000
10
20
30
40
50
dB

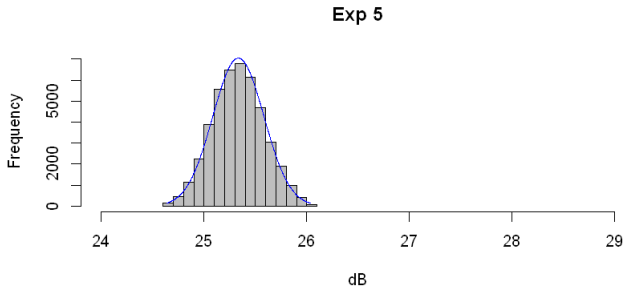

Control
Frequency
0
1000
2500
24
25
26
27
28
29
dB

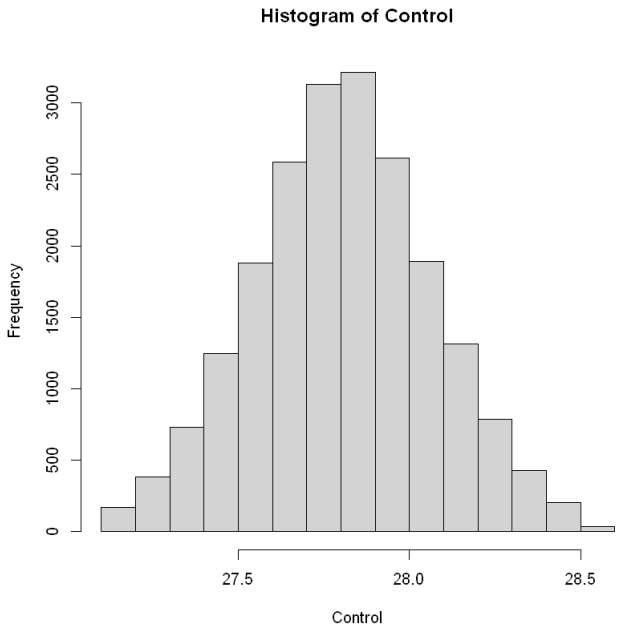

Exp 1
Frequency
0
2000
4000
24
25
26
27
28
29
dB

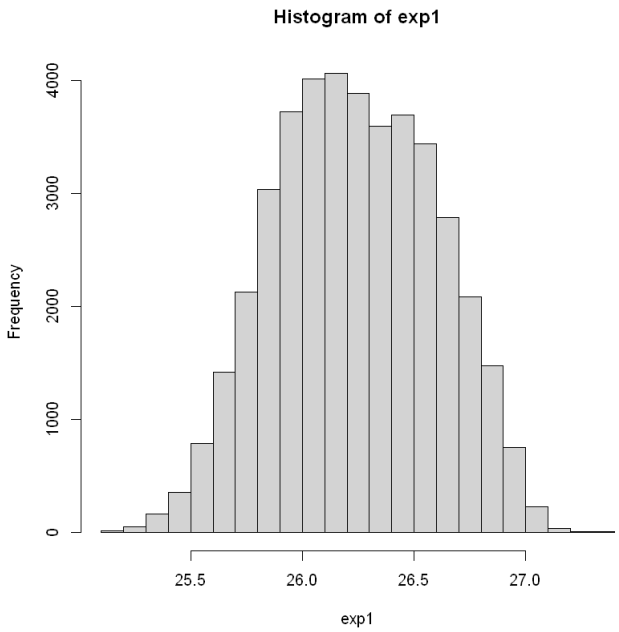

Exp 2
Frequency
0
2000
6000
24
25
26
27
28
29
dB


Exp 3
Frequency
0
2000
5000
24
25
26
27
28
29
dB

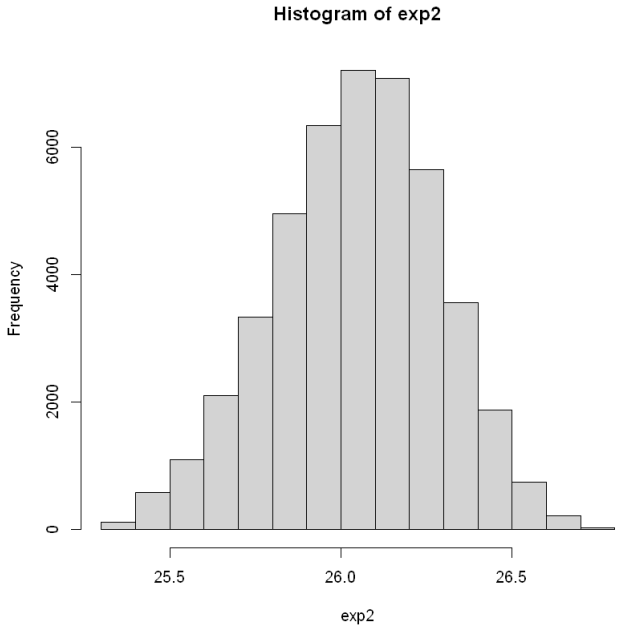

Exp 4
Frequency
0
2000
6000
24
25
26
27
28
29
dB

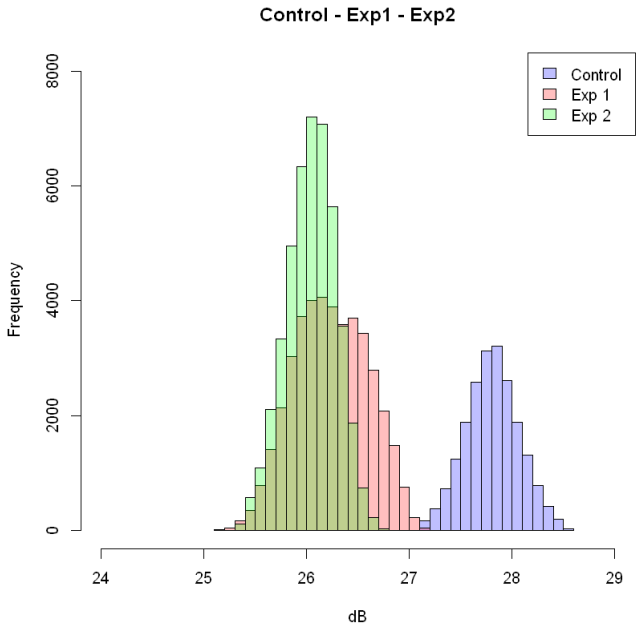

Exp 5
Frequency
0
2000
5000
24
25
26
27
28
29
dB

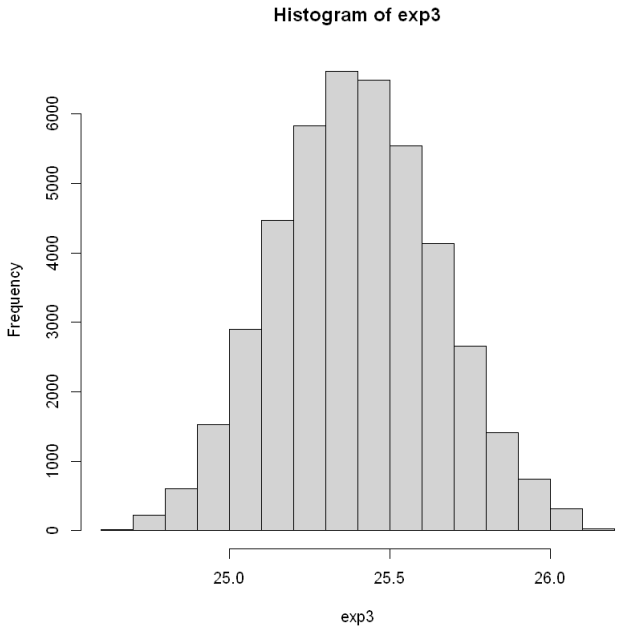

Histogram of Control
Frequency
0
500
1000
1500
2000
2500
3000
27.5
28.0
28.5
Control

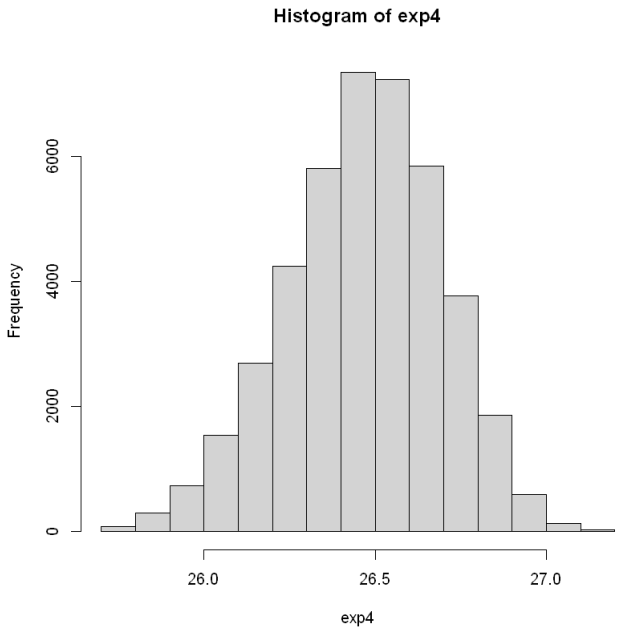

Histogram of exp1
Frequency
0
1000
2000
3000
4000
25.5
26.0
26.5
27.0
exp1

```
Warning message in plot.window(xlim, ylim, "", ...):
""breaks" is not a graphical parameter"
Warning message in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...):
""breaks" is not a graphical parameter"
Warning message in axis(1, ...):
""breaks" is not a graphical parameter"
Warning message in axis(2, at = yt, ...):
""breaks" is not a graphical parameter"
```

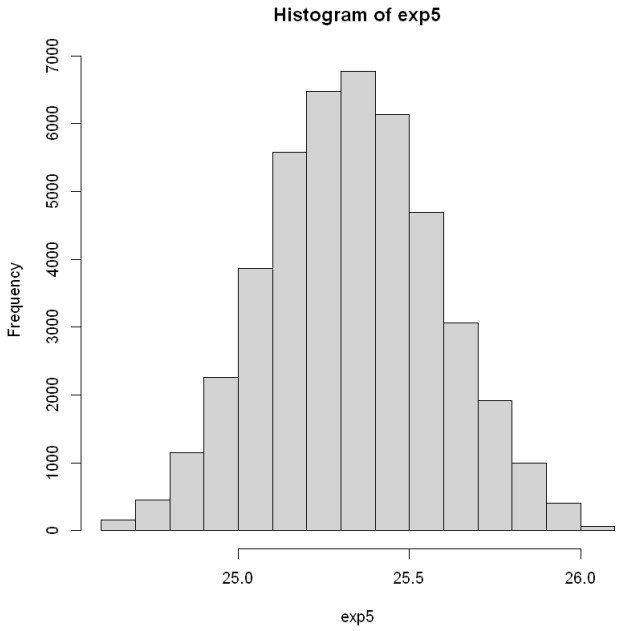


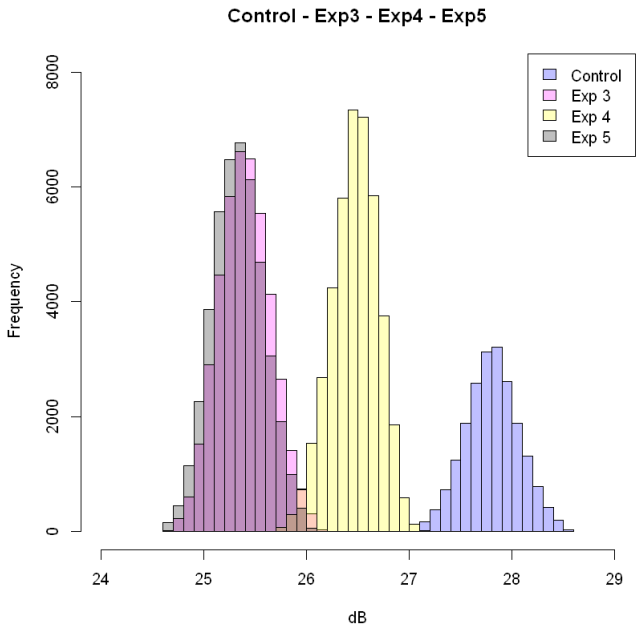







```
Warning message in plot.window(xlim, ylim, "", ...):
""breaks" is not a graphical parameter"
Warning message in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...):
""breaks" is not a graphical parameter"
Warning message in axis(1, ...):
""breaks" is not a graphical parameter"
Warning message in axis(2, at = yt, ...):
""breaks" is not a graphical parameter"
```


Histogram of exp5
Frequency
0
1000
2000
3000
4000
5000
6000
7000
25.0
25.5
26.0
exp5


Control - Exp3 - Exp4 - Exp5
Control
Exp 3
Exp 4
Exp 5
Frequency
0
2000
4000
6000
8000
24
25
26
27
28
29
dB

In [ ]:

In [1]:

```
# t-Test

# 1. Carga de datos

# Librerias
if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(lattice)){install.packages("lattice")}
if(!require(lsr)){install.packages("lsr")}
if(!require(rcompanion)){install.packages("rcompanion")}

# Ingreso de los datos
Datos <- ("
Algoritmo     Ejecucion  Tiempo
'Algoritmo A'   '1'       12070
'Algoritmo A'   '2'       14040
'Algoritmo A'   '3'       13580
'Algoritmo A'   '4'        9540
'Algoritmo A'   '5'       14070
'Algoritmo A'   '6'       11520
'Algoritmo A'   '7'       13030
'Algoritmo A'   '8'       13245
'Algoritmo A'   '9'       14215
'Algoritmo A'   '10'      15070
'Algoritmo A'   '11'      12580
'Algoritmo A'   '12'      11540
'Algoritmo A'   '13'       9580
'Algoritmo A'   '14'      11510
'Algoritmo A'   '15'      16070
'Algoritmo A'   '16'      13010
'Algoritmo A'   '17'      10530
'Algoritmo A'   '18'      13030
'Algoritmo A'   '19'      17080
'Algoritmo A'   '20'      13020
'Algoritmo B'   '1'       11070
'Algoritmo B'   '2'       12010
'Algoritmo B'   '3'       12550
'Algoritmo B'   '4'       10500
'Algoritmo B'   '5'       12000
'Algoritmo B'   '6'       12520
'Algoritmo B'   '7'       13520
'Algoritmo B'   '8'       13540
'Algoritmo B'   '9'       13255
'Algoritmo B'   '10'      15235
'Algoritmo B'   '11'      12235
'Algoritmo B'   '12'      11285
'Algoritmo B'   '13'      10040
'Algoritmo B'   '14'      11295
'Algoritmo B'   '15'      14080
'Algoritmo B'   '16'      12080
'Algoritmo B'   '17'      11580
'Algoritmo B'   '18'      14070
'Algoritmo B'   '19'      15050
'Algoritmo B'   '20'      12050 ")

# 2. Lectura de datos

Data <- read.table(textConnection(Datos), header=TRUE)
rm(Datos)

library(psych)
headTail(Data) # Ordenar datos de mayor a menor

str(Data) # Desplegar de manera compacta
summary(Data) # La estructura del objecto - Verificar que los datos estén correctos

# 3. Resumen organizado

library(FSA)
Summarize(Tiempo ~ Algoritmo, data = Data, digits = 4)


# 4. Análisis de normalidad (histograma + curva normal) Muestras por aparte

# -- Se analiza la normalidad en los datos y se verifica si existe normalidad en los datos.
```

```r
A <- Data$Tiempo[Data$Algoritmo == "Algoritmo A"]
B <- Data$Tiempo[Data$Algoritmo == "Algoritmo B"]

library(rcompanion)
plotNormalHistogram(A)
plotNormalHistogram(B)

# 5. Diagrama de cajas

M <- tapply(Data$Tiempo, INDEX = Data$Algoritmo, FUN = mean)
boxplot(Tiempo ~ Algoritmo, data = Data)
points(M, col = "red", pch = "+", cex = 2)

# 6. Prueba t

t.test(Tiempo ~ Algoritmo, data = Data)
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: lattice

Loading required package: lsr

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 3

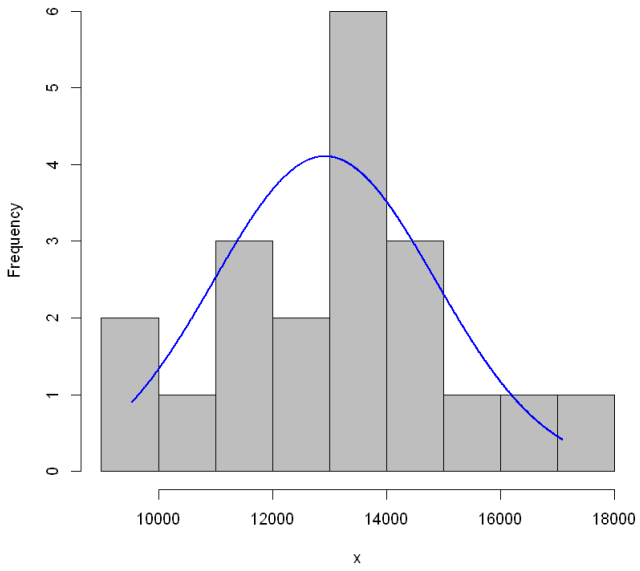
	Algoritmo	Ejecucion	Tiempo
	<chr>	<chr>	<chr>
1	Algoritmo A	1	12070
2	Algoritmo A	2	14040
3	Algoritmo A	3	13580
4	Algoritmo A	4	9540
...	NA	...	...
37	Algoritmo B	17	11580
38	Algoritmo B	18	14070
39	Algoritmo B	19	15050
40	Algoritmo B	20	12050

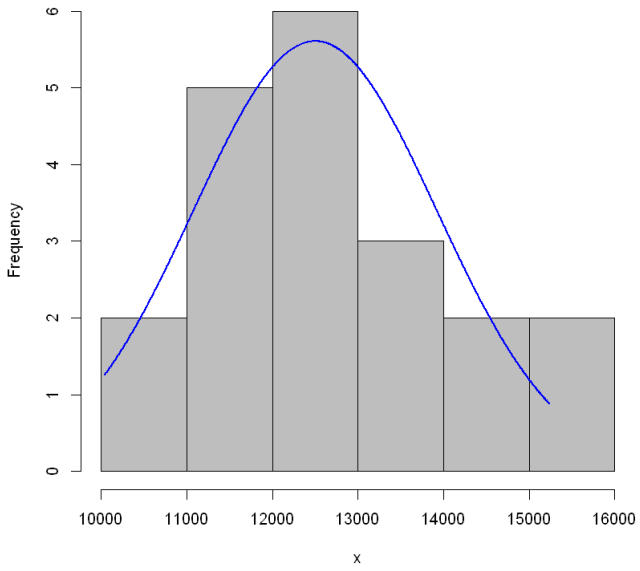
```
'data.frame':   40 obs. of  3 variables:
 $ Algoritmo: chr  "Algoritmo A" "Algoritmo A" "Algoritmo A" "Algoritmo A" ...
 $ Ejecucion: int  1 2 3 4 5 6 7 8 9 10 ...
 $ Tiempo   : int  12070 14040 13580 9540 14070 11520 13030 13245 14215 15070 ...
```

```
 Algoritmo           Ejecucion         Tiempo
Length:40          Min.   : 1.00   Min.   : 9540
Class :character   1st Qu.: 5.75   1st Qu.:11535
Mode  :character   Median :10.50   Median :12565
                   Mean   :10.50   Mean   :12707
                   3rd Qu.:15.25   3rd Qu.:13695
                   Max.   :20.00   Max.   :17080
```

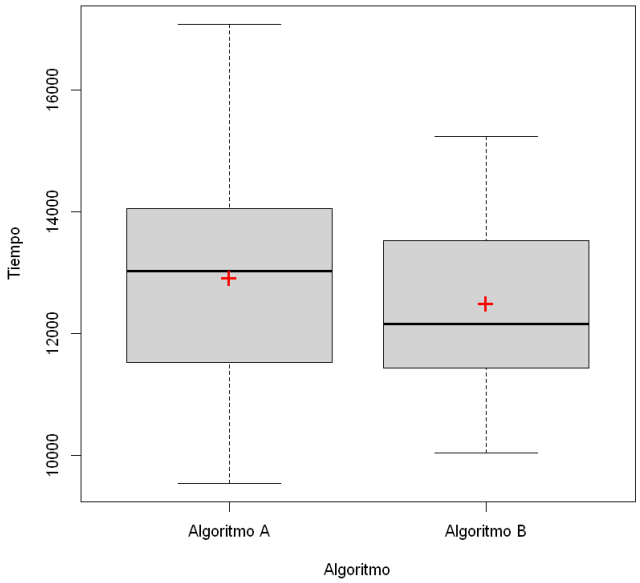
A data.frame: 2 × 9

Algoritmo	n	mean	sd	min	Q1	median	Q3	max
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	20	12916.50	1942.198	9540	11535.00	13025.0	14047.5	17080
Algoritmo B	20	12498.25	1422.549	10040	11508.75	12157.5	13525.0	15235






```
        Welch Two Sample t-test

data:  Tiempo by Algoritmo
t = 0.77695, df = 34.83, p-value = 0.4424
alternative hypothesis: true difference in means between group Algoritmo A and group Algoritmo B is not equ
al to 0
95 percent confidence interval:
 -674.7892 1511.2892
sample estimates:
mean in group Algoritmo A mean in group Algoritmo B
                 12916.50                  12498.25
```


Tiempo
16000
14000
12000
10000
Algoritmo A
Algoritmo B
Algoritmo

In [ ]:

In [1]:

```
# Anova diseño multifactorial 2^k

# 1. Carga inicial de datos.

if(!require(psych)){install.packages("psych")}
if(!require(FSA)){install.packages("FSA")}
if(!require(ggplot2)){install.packages("ggplot2")}
if(!require(car)){install.packages("car")}
if(!require(multcompView)){install.packages("multcompView")}
if(!require(lsmeans)){install.packages("lsmeans")}
if(!require(rcompanion)){install.packages("rcompanion")}


ln <- ("
Algoritmo       Entrenamiento       Rendimiento   Acelerador
'Algoritmo A'        MT500           12000           NA-NA
'Algoritmo A'        MT500           14005           NA-NA
'Algoritmo A'        MT500           13508           NA-NA
'Algoritmo A'        MT500            9503           NA-NA
'Algoritmo A'        MT500           14004           NA-NA
'Algoritmo A'        MT1000          11502           NA-NA
'Algoritmo A'        MT1000          13006           NA-NA
'Algoritmo A'        MT1000          13252           NA-NA
'Algoritmo A'        MT1000          14253           NA-NA
'Algoritmo A'        MT1000          15003           NA-NA
'Algoritmo A'        MT5000          12504           NA-NA
'Algoritmo A'        MT5000          11504           NA-NA
'Algoritmo A'        MT5000           9500           NA-NA
'Algoritmo A'        MT5000          11506           NA-NA
'Algoritmo A'        MT5000          16000           NA-NA
'Algoritmo A'        MT50000         13008           NA-NA
'Algoritmo A'        MT50000         10506           NA-NA
'Algoritmo A'        MT50000         13005           NA-NA
'Algoritmo A'        MT50000         17002           NA-NA
'Algoritmo A'        MT50000         13008           NA-NA
'Algoritmo B'        MT500           11005           NA-NA
'Algoritmo B'        MT500           12007           NA-NA
'Algoritmo B'        MT500           12509           NA-NA
'Algoritmo B'        MT500           10504           NA-NA
'Algoritmo B'        MT500           12002           NA-NA
'Algoritmo B'        MT1000          12504           NA-NA
'Algoritmo B'        MT1000          13501           NA-NA
'Algoritmo B'        MT1000          13501           NA-NA
'Algoritmo B'        MT1000          13252           NA-NA
'Algoritmo B'        MT1000          15256           NA-NA
'Algoritmo B'        MT5000          12253           NA-NA
'Algoritmo B'        MT5000          11255           NA-NA
'Algoritmo B'        MT5000          10006           NA-NA
'Algoritmo B'        MT5000          11252           NA-NA
'Algoritmo B'        MT5000          14004           NA-NA
'Algoritmo B'        MT50000         12007           NA-NA
'Algoritmo B'        MT50000         11505           NA-NA
'Algoritmo B'        MT50000         14009           NA-NA
'Algoritmo B'        MT50000         15000           NA-NA
'Algoritmo B'        MT50000         12009           NA-NA
'Algoritmo C'        MT500            9000           NA-NA
'Algoritmo C'        MT500           11003           NA-NA
'Algoritmo C'        MT500           11505           NA-NA
'Algoritmo C'        MT500            9509           NA-NA
'Algoritmo C'        MT500           11003           NA-NA
'Algoritmo C'        MT1000          11508           NA-NA
'Algoritmo C'        MT1000          12508           NA-NA
'Algoritmo C'        MT1000          12506           NA-NA
'Algoritmo C'        MT1000          12254           NA-NA
'Algoritmo C'        MT1000          13253           NA-NA
'Algoritmo C'        MT5000          11255           NA-NA
'Algoritmo C'        MT5000          10257           NA-NA
'Algoritmo C'        MT5000           9500           NA-NA
'Algoritmo C'        MT5000           9255           NA-NA
'Algoritmo C'        MT5000          12009           NA-NA
'Algoritmo C'        MT50000         11000           NA-NA
'Algoritmo C'        MT50000          9509           NA-NA
'Algoritmo C'        MT50000         13009           NA-NA
'Algoritmo C'        MT50000         14005           NA-NA
'Algoritmo C'        MT50000         11001           NA-NA
```

```
'Algoritmo A'       MT500          12046          NA-SW
'Algoritmo A'       MT500          14589          NA-SW
'Algoritmo A'       MT500          13723          NA-SW
'Algoritmo A'       MT500           9799          NA-SW
'Algoritmo A'       MT500          14715          NA-SW
'Algoritmo A'       MT1000         11144          NA-SW
'Algoritmo A'       MT1000         13920          NA-SW
'Algoritmo A'       MT1000         13226          NA-SW
'Algoritmo A'       MT1000         14845          NA-SW
'Algoritmo A'       MT1000         15142          NA-SW
'Algoritmo A'       MT5000         12352          NA-SW
'Algoritmo A'       MT5000         11296          NA-SW
'Algoritmo A'       MT5000          9737          NA-SW
'Algoritmo A'       MT5000         11129          NA-SW
'Algoritmo A'       MT5000         16409          NA-SW
'Algoritmo A'       MT50000        13872          NA-SW
'Algoritmo A'       MT50000        10100          NA-SW
'Algoritmo A'       MT50000        13419          NA-SW
'Algoritmo A'       MT50000        17398          NA-SW
'Algoritmo A'       MT50000        13164          NA-SW
'Algoritmo B'       MT500          11047          NA-SW
'Algoritmo B'       MT500          12226          NA-SW
'Algoritmo B'       MT500          12105          NA-SW
'Algoritmo B'       MT500          10418          NA-SW
'Algoritmo B'       MT500          12446          NA-SW
'Algoritmo B'       MT1000         12156          NA-SW
'Algoritmo B'       MT1000         13968          NA-SW
'Algoritmo B'       MT1000         13891          NA-SW
'Algoritmo B'       MT1000         13778          NA-SW
'Algoritmo B'       MT1000         15448          NA-SW
'Algoritmo B'       MT5000         12441          NA-SW
'Algoritmo B'       MT5000         11767          NA-SW
'Algoritmo B'       MT5000         10340          NA-SW
'Algoritmo B'       MT5000         11306          NA-SW
'Algoritmo B'       MT5000         14565          NA-SW
'Algoritmo B'       MT50000        12725          NA-SW
'Algoritmo B'       MT50000        11169          NA-SW
'Algoritmo B'       MT50000        14749          NA-SW
'Algoritmo B'       MT50000        15566          NA-SW
'Algoritmo B'       MT50000        12239          NA-SW
'Algoritmo C'       MT500           9082          NA-SW
'Algoritmo C'       MT500          11887          NA-SW
'Algoritmo C'       MT500          11799          NA-SW
'Algoritmo C'       MT500           9300          NA-SW
'Algoritmo C'       MT500          11049          NA-SW
'Algoritmo C'       MT1000         11378          NA-SW
'Algoritmo C'       MT1000         12659          NA-SW
'Algoritmo C'       MT1000         12905          NA-SW
'Algoritmo C'       MT1000         12782          NA-SW
'Algoritmo C'       MT1000         13196          NA-SW
'Algoritmo C'       MT5000         11795          NA-SW
'Algoritmo C'       MT5000         10316          NA-SW
'Algoritmo C'       MT5000          9947          NA-SW
'Algoritmo C'       MT5000          9420          NA-SW
'Algoritmo C'       MT5000         12699          NA-SW
'Algoritmo C'       MT50000        11024          NA-SW
'Algoritmo C'       MT50000         9556          NA-SW
'Algoritmo C'       MT50000        13900          NA-SW
'Algoritmo C'       MT50000        14006          NA-SW
'Algoritmo C'       MT50000        11738          NA-SW
'Algoritmo A'       MT500          126572         HW-NA
'Algoritmo A'       MT500          140058         HW-NA
'Algoritmo A'       MT500          139580         HW-NA
'Algoritmo A'       MT500           92583         HW-NA
'Algoritmo A'       MT500          148057         HW-NA
'Algoritmo A'       MT1000         110078         HW-NA
'Algoritmo A'       MT1000         131942         HW-NA
'Algoritmo A'       MT1000         133797         HW-NA
'Algoritmo A'       MT1000         140026         HW-NA
'Algoritmo A'       MT1000         155479         HW-NA
'Algoritmo A'       MT5000         125809         HW-NA
'Algoritmo A'       MT5000         114264         HW-NA
'Algoritmo A'       MT5000          98797         HW-NA
'Algoritmo A'       MT5000         113400         HW-NA
'Algoritmo A'       MT5000         168898         HW-NA
'Algoritmo A'       MT50000        133452         HW-NA
```

```
'Algoritmo A'       MT50000        101641         HW-NA
'Algoritmo A'       MT50000        133155         HW-NA
'Algoritmo A'       MT50000        175156         HW-NA
'Algoritmo A'       MT50000        131945         HW-NA
'Algoritmo B'       MT500          110317         HW-NA
'Algoritmo B'       MT500          129244         HW-NA
'Algoritmo B'       MT500          127966         HW-NA
'Algoritmo B'       MT500          109783         HW-NA
'Algoritmo B'       MT500          122936         HW-NA
'Algoritmo B'       MT1000         128830         HW-NA
'Algoritmo B'       MT1000         134437         HW-NA
'Algoritmo B'       MT1000         138321         HW-NA
'Algoritmo B'       MT1000         132000         HW-NA
'Algoritmo B'       MT1000         157693         HW-NA
'Algoritmo B'       MT5000         121964         HW-NA
'Algoritmo B'       MT5000         119872         HW-NA
'Algoritmo B'       MT5000         106654         HW-NA
'Algoritmo B'       MT5000         112666         HW-NA
'Algoritmo B'       MT5000         145535         HW-NA
'Algoritmo B'       MT50000        127938         HW-NA
'Algoritmo B'       MT50000        115179         HW-NA
'Algoritmo B'       MT50000        143021         HW-NA
'Algoritmo B'       MT50000        150357         HW-NA
'Algoritmo B'       MT50000        121216         HW-NA
'Algoritmo C'       MT500           95474         HW-NA
'Algoritmo C'       MT500          113776         HW-NA
'Algoritmo C'       MT500          117473         HW-NA
'Algoritmo C'       MT500           92900         HW-NA
'Algoritmo C'       MT500          115582         HW-NA
'Algoritmo C'       MT1000         115279         HW-NA
'Algoritmo C'       MT1000         122184         HW-NA
'Algoritmo C'       MT1000         124770         HW-NA
'Algoritmo C'       MT1000         128403         HW-NA
'Algoritmo C'       MT1000         135219         HW-NA
'Algoritmo C'       MT5000         112562         HW-NA
'Algoritmo C'       MT5000         108736         HW-NA
'Algoritmo C'       MT5000          91064         HW-NA
'Algoritmo C'       MT5000          98171         HW-NA
'Algoritmo C'       MT5000         120277         HW-NA
'Algoritmo C'       MT50000        111299         HW-NA
'Algoritmo C'       MT50000         90193         HW-NA
'Algoritmo C'       MT50000        135178         HW-NA
'Algoritmo C'       MT50000        146158         HW-NA
'Algoritmo C'       MT50000        113845         HW-NA
'Algoritmo A'       MT500          124252         HW-SW
'Algoritmo A'       MT500          143833         HW-SW
'Algoritmo A'       MT500          138907         HW-SW
'Algoritmo A'       MT500           91010         HW-SW
'Algoritmo A'       MT500          143901         HW-SW
'Algoritmo A'       MT1000         116563         HW-SW
'Algoritmo A'       MT1000         136455         HW-SW
'Algoritmo A'       MT1000         130411         HW-SW
'Algoritmo A'       MT1000         140060         HW-SW
'Algoritmo A'       MT1000         154308         HW-SW
'Algoritmo A'       MT5000         124480         HW-SW
'Algoritmo A'       MT5000         111552         HW-SW
'Algoritmo A'       MT5000          99135         HW-SW
'Algoritmo A'       MT5000         110208         HW-SW
'Algoritmo A'       MT5000         167228         HW-SW
'Algoritmo A'       MT50000        134267         HW-SW
'Algoritmo A'       MT50000        102119         HW-SW
'Algoritmo A'       MT50000        138036         HW-SW
'Algoritmo A'       MT50000        171632         HW-SW
'Algoritmo A'       MT50000        130666         HW-SW
'Algoritmo B'       MT500          116942         HW-SW
'Algoritmo B'       MT500          129721         HW-SW
'Algoritmo B'       MT500          128834         HW-SW
'Algoritmo B'       MT500          100390         HW-SW
'Algoritmo B'       MT500          127771         HW-SW
'Algoritmo B'       MT1000         121789         HW-SW
'Algoritmo B'       MT1000         135311         HW-SW
'Algoritmo B'       MT1000         136587         HW-SW
'Algoritmo B'       MT1000         139664         HW-SW
'Algoritmo B'       MT1000         151543         HW-SW
'Algoritmo B'       MT5000         128962         HW-SW
'Algoritmo B'       MT5000         110157         HW-SW
```

```
'Algoritmo B'        MT5000          106129          HW-SW
'Algoritmo B'        MT5000          114634          HW-SW
'Algoritmo B'        MT5000          143337          HW-SW
'Algoritmo B'        MT50000         129292          HW-SW
'Algoritmo B'        MT50000         117502          HW-SW
'Algoritmo B'        MT50000         143687          HW-SW
'Algoritmo B'        MT50000         153488          HW-SW
'Algoritmo B'        MT50000         129773          HW-SW
'Algoritmo C'        MT500            99920          HW-SW
'Algoritmo C'        MT500           110833          HW-SW
'Algoritmo C'        MT500           117879          HW-SW
'Algoritmo C'        MT500            96441          HW-SW
'Algoritmo C'        MT500           119688          HW-SW
'Algoritmo C'        MT1000          117995          HW-SW
'Algoritmo C'        MT1000          122984          HW-SW
'Algoritmo C'        MT1000          120317          HW-SW
'Algoritmo C'        MT1000          120213          HW-SW
'Algoritmo C'        MT1000          137806          HW-SW
'Algoritmo C'        MT5000          117014          HW-SW
'Algoritmo C'        MT5000          105529          HW-SW
'Algoritmo C'        MT5000           98755          HW-SW
'Algoritmo C'        MT5000           96010          HW-SW
'Algoritmo C'        MT5000          126548          HW-SW
'Algoritmo C'        MT50000         113527          HW-SW
'Algoritmo C'        MT50000          99385          HW-SW
'Algoritmo C'        MT50000         136573          HW-SW
'Algoritmo C'        MT50000         141965          HW-SW
'Algoritmo C'        MT50000         111994          HW-SW
")


# Se introduce la tabla.
Data <- read.table(textConnection(ln), header=TRUE)

# Se ordenan los datos según los ingresamos. (Evitar orden alfabético por R).
Data$Entrenamiento <- factor(Data$Entrenamiento, levels = unique(Data$Entrenamiento))
Data$Acelerador <- factor(Data$Acelerador, levels = unique(Data$Acelerador))
Data$Algoritmo <- factor(Data$Algoritmo, levels = unique(Data$Algoritmo))

# 2. Verificación de la lectura de datos.

library(psych)
headTail(Data)
str(Data)
summary(Data)
rm(ln)

# 3. Gráfico simple de interacción.

# Variable dependiente: Rendimiento
# Variables independientes: Algoritmo y Método de Entrenamiento.
interaction.plot(x.factor = Data$Entrenamiento,
 trace.factor = Data$Algoritmo,
 response = Data$Rendimiento,
 fun = mean,
 type = "b",
 col = c("black", "red", "green"),
 pch = c(19,17,15),
 fixed = TRUE,
 leg.bty = "o")


# 4. Se realiza cambio al gráfico para agregar el acelerador.
# Algoritmo en función del acelerador para ver el rendimiento.

interaction.plot(x.factor = Data$Acelerador,
                 trace.factor = Data$Algoritmo,
                 response = Data$Rendimiento,
                 fun = mean,
                 type = "b",
                 col = c("black", "red", "green"),
                 pch = c(19, 17, 15),
                 fixed = TRUE,
                 leg.bty = "o")

# 5. Modelo lineal y anova
```

```r
# * Analisis de factores e interaccinoes de los factores.
model <- lm(Rendimiento ~ Entrenamiento * Algoritmo * Acelerador, data = Data)
library(car)
Anova(model, type = "II")

# Hay diferencias entre los grupos de entrenamiento, los algoritmos y el acelerador.
# Las factores impactan la variable de  respuesta.
# Las interacciones no impactan las variables de respuesta | Algoritmo:Acelerador

# 6. Evaluacion de los supuestos

x <- residuals(model)
library(rcompanion)
plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)

# Parece haber normalidad, no hay homocedasticidad.

# Se procede a hacer transformación iniciando desde la forma menos agresiva a la mas agresiva, hasta cumplir

# 7. Transformación por raiz cuadrada

library(rcompanion)
T_sqrt <- sqrt(Data$Rendimiento) # Ingresar variable dependiente.
model <- lm(T_sqrt ~ Entrenamiento * Algoritmo * Acelerador, data = Data)

library(car)
Anova(model, type = "II")

# Supuestos

x <- residuals(model)
library(rcompanion)
plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)

# Estamos llegando a conclusiones no al rendimiento, sino a la raiz cuadrada del rendimiento, porque eso es


# 8. Transformación por raíz cúbica

library(rcompanion)
 T_cub <- sign(Data$Rendimiento) * abs(Data$Rendimiento)^(1/3) # Ingresar variable dependiente.
model <- lm(T_cub ~ Entrenamiento * Algoritmo * Acelerador, data = Data)

library(car)
Anova(model, type = "II")

# Supuestos

x <- residuals(model)
library(rcompanion)
plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)


# 9. Transformación por logaritmo

library(rcompanion)
T_log <- log(Data$Rendimiento)

model <- lm(T_log ~ Entrenamiento * Algoritmo * Acelerador, data = Data)
library(car)
Anova(model, type = "II")

x <- residuals(model)
library(rcompanion)

plotNormalHistogram(x)
plot(fitted(model), residuals(model))
plot(model)

# 10. Prueba Levene
```

```r
leveneTest(T_log ~ Entrenamiento * Algoritmo * Acelerador, data = Data)

# 11. Analisis post-hoc por algoritmo

library(lsmeans)
marginal <- lsmeans(model, pairwise ~ Algoritmo, adjust="tukey")

library(multcomp)
CLD <- cld(marginal, alpha = 0.05, Letters = letters, adjust = "tukey")
CLD

# 12. Analisis post-hoc por entrenamiento

library(lsmeans)
marginal <- lsmeans(model, pairwise ~ Entrenamiento, adjust="tukey")

library(multcomp)
CLD <- cld(marginal, alpha = 0.05, Letters = letters, adjunst = "tukey")
CLD

# 13. Análisis post-hoc para acelerador 2^k

library(lsmeans)
marginal <- lsmeans(model, pairwise ~ Acelerador, adjust="tukey")

library(multcomp)
CLD <- cld(marginal, alpha = 0.05, Letters = letters, adjunst = "tukey")
CLD

# 14. Gráficos finales.

library(FSA)
Sum <- Summarize(T_log ~ Entrenamiento + Algoritmo, data = Data, digits = 3)

# Se agrega el se
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum

### Ordenamos

Sum$Entrenamiento <- factor(Sum$Entrenamiento, levels = unique(Sum$Entrenamiento))

### Graficamos

library(ggplot2)
pd <- position_dodge(.2)

ggplot(Sum, aes(x = Entrenamiento,
                y = mean, color = Algoritmo)) +
    geom_errorbar(aes(ymin = mean - se, ymax = mean + se), width = .2, size = 0.7, position = pd) +
    geom_point(shape = 15, size = 4, position = pd) +
    theme_bw() +
    theme(axis.title = element_text(face = "bold")) +
    scale_colour_manual(values = c("black", "red", "green")) +
    ylab("Logaritmo de rendimiento")


# Para acelerador

Sum <- Summarize(T_log ~ Acelerador + Algoritmo, data = Data, digits = 3)

# Se agrega el se
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum

### Ordenamos | Correccion: se cambia entrenamiento por acelerador
Sum$Acelerador <- factor(Sum$Acelerador, levels = unique(Sum$Acelerador))

### Graficamos

ggplot(Sum, aes(x = Acelerador,
                y = mean, color = Algoritmo)) +
    geom_errorbar(aes(ymin = mean - se, ymax = mean + se), width = .2, size = 0.7, position = pd) +
```

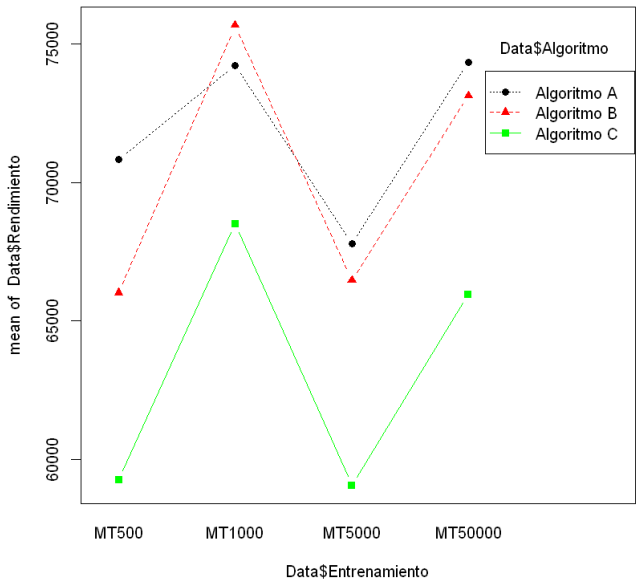
```r
    geom_point(shape = 15, size = 4, position = pd) +
    theme_bw() +
    theme(axis.title = element_text(face = "bold")) +
    scale_colour_manual(values = c("black", "red", "green")) +
    ylab("Logaritmo de rendimiento")

# Bigotes pequeños por ser dato transformados


# 15. Gráfico de promedios transformados

### Creamos un dato llamado sum con promedios y se
library(FSA)
Sum <- Summarize(T_log ~ Algoritmo, data = Data, digits = 3)

### Agregamos el se
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum

### Ordenamos | Correccion: se cambia entrenamiento por algoritmo
Sum$Algoritmo <- factor(Sum$Algoritmo, levels = unique(Sum$Algoritmo))

### Graficamos
library(ggplot2)
pd <- position_dodge(.2)

# Correccion: se cambia entrenamiento por algoritmo
ggplot(Sum, aes(x = Algoritmo,
                y = mean, color = Algoritmo)) +
    geom_errorbar(aes(ymin = mean - se, ymax = mean + se), width = .2, size = 0.7, position = pd) +
    geom_point(shape = 15, size = 4, position = pd) +
    theme_bw() +
    theme(axis.title = element_text(face = "bold")) +
    scale_colour_manual(values = c("black", "red", "green")) +
    ylab("Logaritmo de rendimiento")

# 16. Des-transformando promedios

library(FSA)
# Corrección, no se agrega Entrenamiento. El grafico final no seria el mismo en la presentacion.
Sum <- Summarize(T_log ~ Algoritmo, data = Data, digits = 3)

Sum$mean <- exp(Sum$mean)
Sum$sd <- exp(Sum$sd)

### Agregamos el se
Sum$se <- Sum$sd / sqrt(Sum$n)
Sum$se <- signif(Sum$se, digits = 3)
Sum

ggplot(Sum, aes(x = Algoritmo,
                y = mean, color = Algoritmo)) +
    geom_errorbar(aes(ymin = mean - se, ymax = mean + se), width = .2, size = 0.7, position = pd) +
    geom_point(shape = 15, size = 4, position = pd) +
    theme_bw() +
    theme(axis.title = element_text(face = "bold")) +
    scale_colour_manual(values = c("black", "red", "green")) +
    ylab("Logaritmo de rendimiento")
```

```
Loading required package: psych

Loading required package: FSA

## FSA v0.9.4. See citation('FSA') if used in publication.
## Run fishR() for related website and fishR('IFAR') for related book.


Attaching package: 'FSA'


The following object is masked from 'package:psych':

    headtail


Loading required package: ggplot2


Attaching package: 'ggplot2'


The following objects are masked from 'package:psych':

    %+%, alpha


Loading required package: car

Loading required package: carData

Registered S3 methods overwritten by 'car':
  method       from
  hist.boot    FSA
  confint.boot FSA


Attaching package: 'car'


The following object is masked from 'package:FSA':

    bootCase


The following object is masked from 'package:psych':

    logit


Loading required package: multcompView

Loading required package: lsmeans

Loading required package: emmeans

The 'lsmeans' package is now basically a front end for 'emmeans'.
Users are encouraged to switch the rest of the way.
See help('transition') for more information, including how to
convert old 'lsmeans' objects and scripts to work with 'emmeans'.

Loading required package: rcompanion


Attaching package: 'rcompanion'


The following object is masked from 'package:psych':

    phi
```

A data.frame: 9 × 4

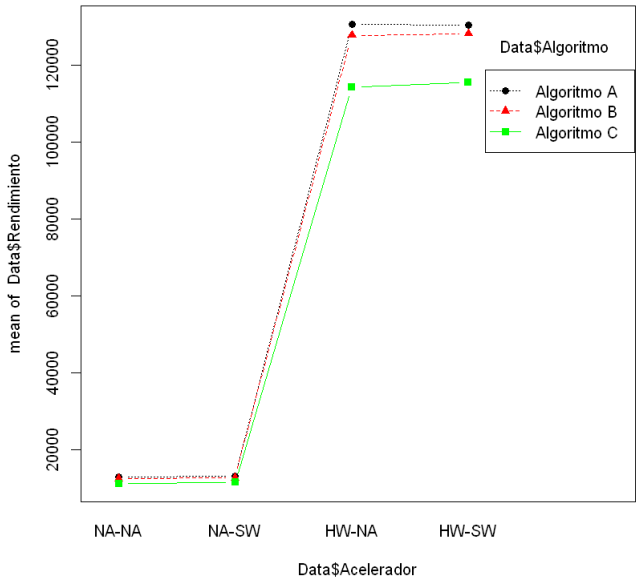
	Algoritmo	Entrenamiento	Rendimiento	Acelerador
	<fct>	<fct>	<chr>	<fct>
1	Algoritmo A	MT500	12000	NA-NA
2	Algoritmo A	MT500	14005	NA-NA
3	Algoritmo A	MT500	13508	NA-NA
4	Algoritmo A	MT500	9503	NA-NA
...	NA	NA	...	NA
237	Algoritmo C	MT50000	99385	HW-SW
238	Algoritmo C	MT50000	136573	HW-SW
239	Algoritmo C	MT50000	141965	HW-SW
240	Algoritmo C	MT50000	111994	HW-SW

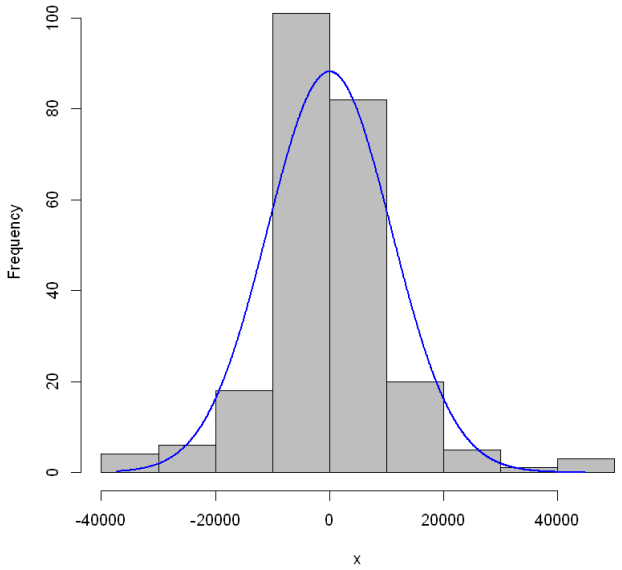
```
'data.frame':   240 obs. of  4 variables:
 $ Algoritmo    : Factor w/ 3 levels "Algoritmo A",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Entrenamiento: Factor w/ 4 levels "MT500","MT1000",..: 1 1 1 1 2 2 2 2 2 ...
 $ Rendimiento  : int  12000 14005 13508 9503 14004 11502 13006 13252 14253 15003 ...
 $ Acelerador   : Factor w/ 4 levels "NA-NA","NA-SW",..: 1 1 1 1 1 1 1 1 1 1 ...
       Algoritmo  Entrenamiento  Rendimiento     Acelerador
 Algoritmo A:80   MT500  :60    Min.   :  9000   NA-NA:60
 Algoritmo B:80   MT1000 :60    1st Qu.: 12236   NA-SW:60
 Algoritmo C:80   MT5000 :60    Median : 53796   HW-NA:60
                  MT50000:60    Mean   : 68432   HW-SW:60
                                3rd Qu.:124309
                                Max.   :175156
```

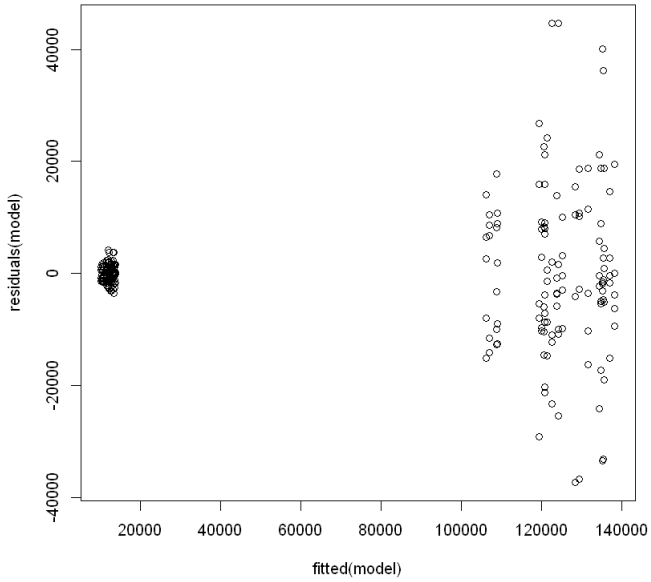


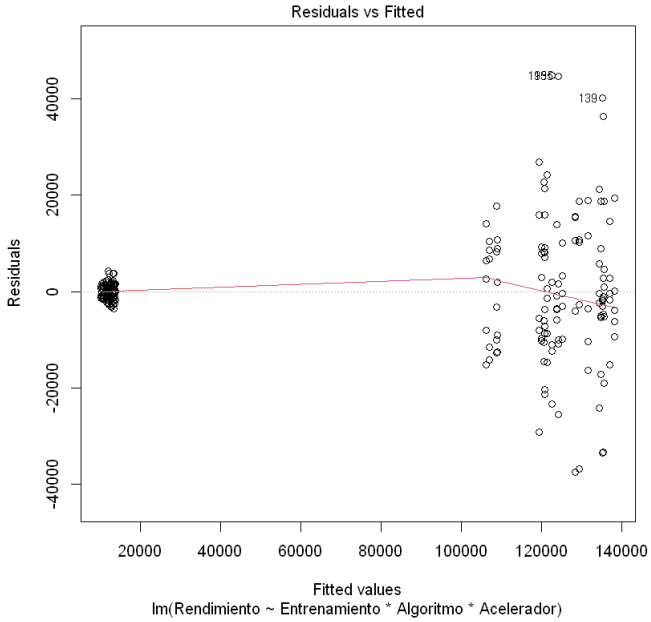
A anova: 8 × 4

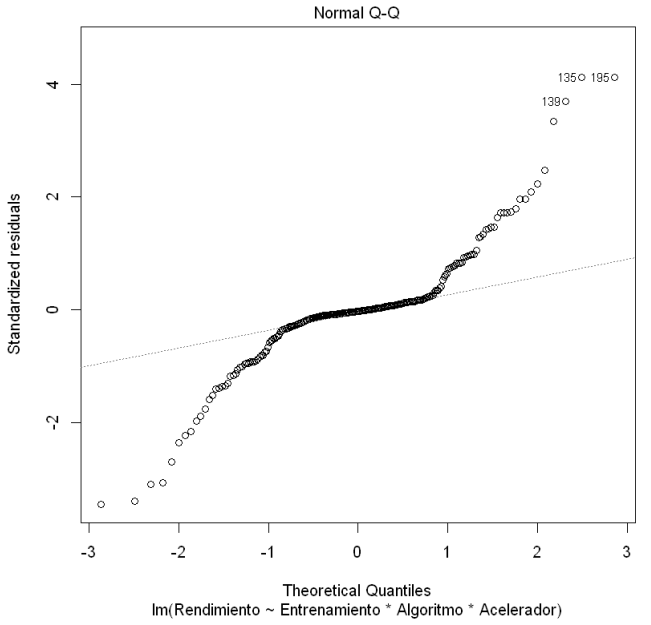
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Entrenamiento	3105643182	3	7.049149e+00	1.607907e-04
Algoritmo	3384497412	2	1.152313e+01	1.878690e-05
Acelerador	755617765537	3	1.715091e+03	2.580204e-138
Entrenamiento:Algoritmo	248909753	6	2.824861e-01	9.447359e-01
Entrenamiento:Acelerador	2002310059	9	1.514939e+00	1.449261e-01
Algoritmo:Acelerador	2278154237	6	2.585462e+00	1.970571e-02
Entrenamiento:Algoritmo:Acelerador	202043559	18	7.643264e-02	9.999999e-01
Residuals	28196478254	192	NA	NA

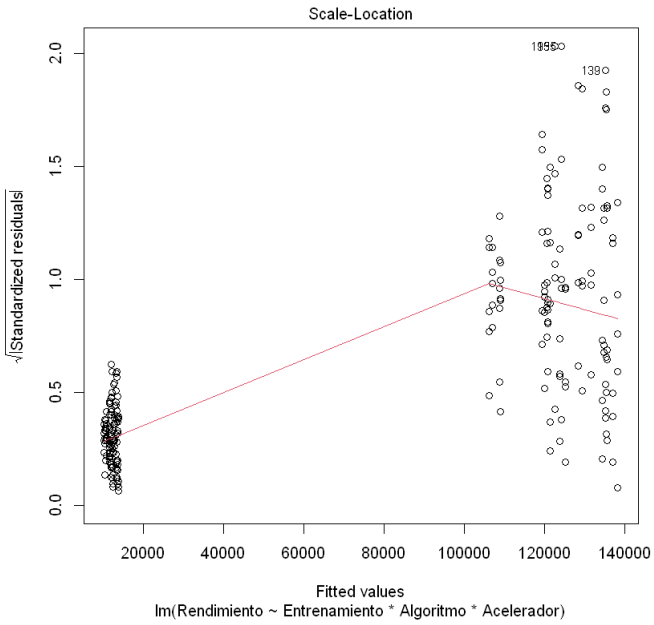

Data$Algoritmo
Algoritmo A
Algoritmo B
Algoritmo C
mean of Data$Rendimiento
20000
40000
60000
80000
100000
120000
NA-NA
NA-SW
HW-NA
HW-SW
Data$Acelerador


Frequency
0
20
40
60
80
100
-40000
-20000
0
20000
40000
x


residuals(model)
-40000
-20000
0
20000
40000
20000
40000
60000
80000
100000
120000
140000
fitted(model)

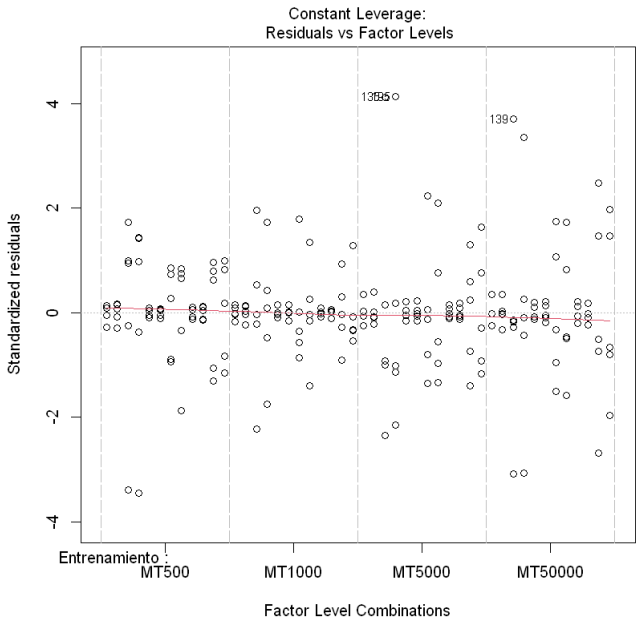

Residuals vs Fitted
Residuals
Fitted values
lm(Rendimiento ~ Entrenamiento * Algoritmo * Acelerador)

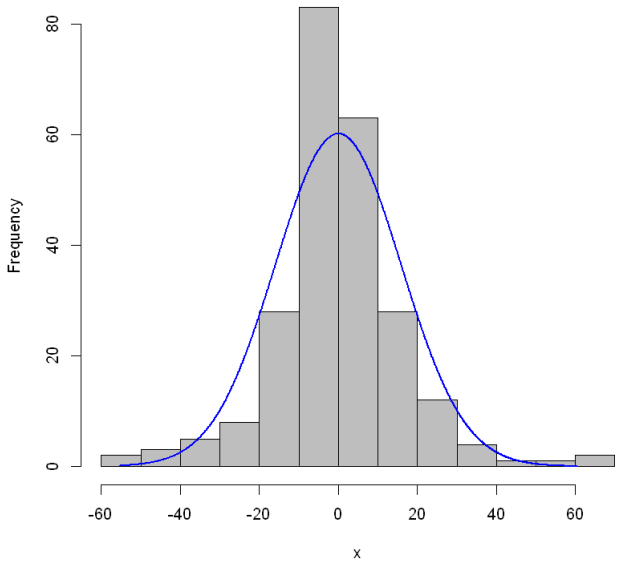

Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Rendimiento ~ Entrenamiento * Algoritmo * Acelerador)

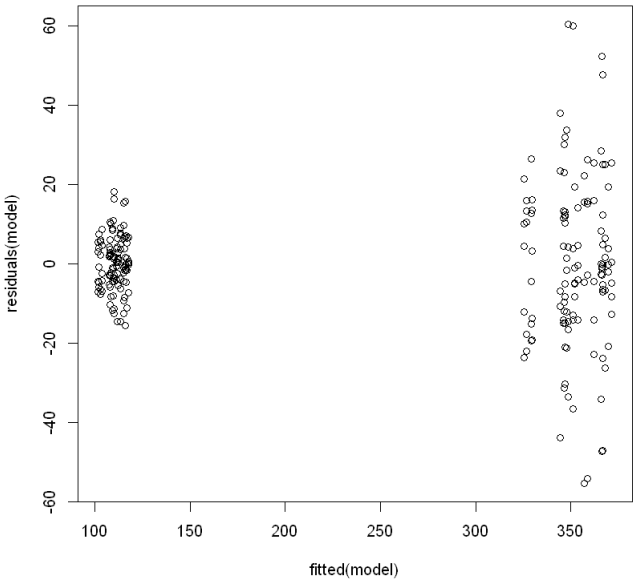

Scale-Location
√|Standardized residuals|
Fitted values
lm(Rendimiento ~ Entrenamiento * Algoritmo * Acelerador)

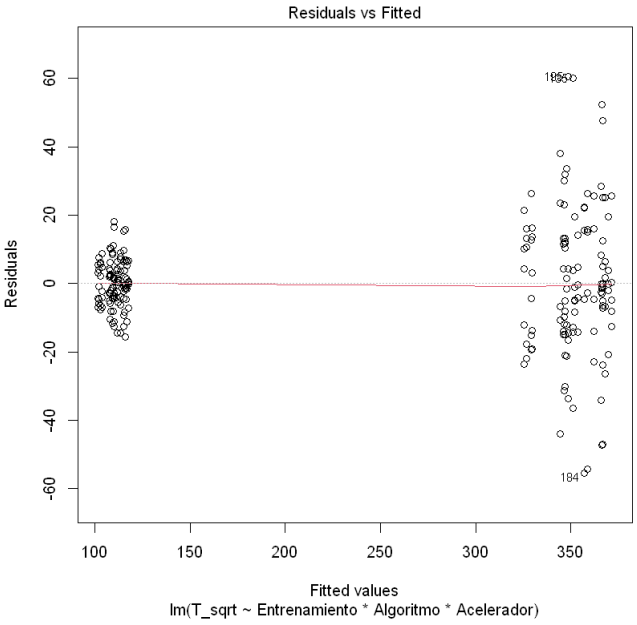
A anova: 8 × 4

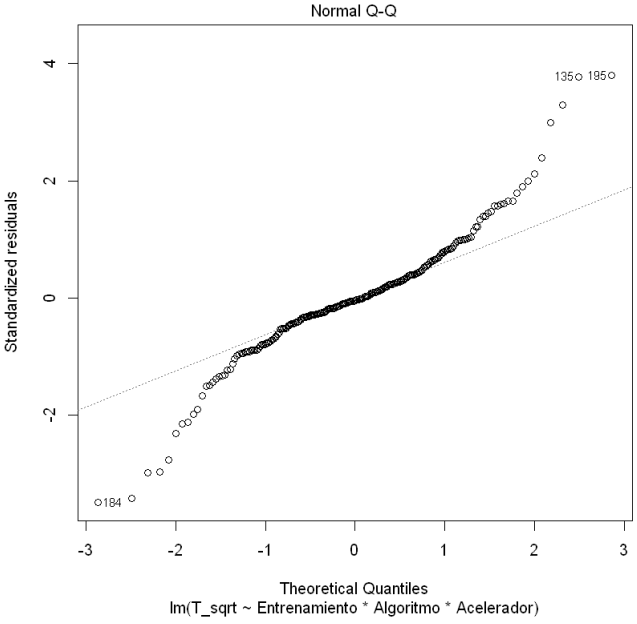
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Entrenamiento	9545.0129	3	1.008357e+01	3.355619e-06
Algoritmo	9671.1896	2	1.532531e+01	6.683816e-07
Acelerador	3492852.0776	3	3.689930e+03	1.923920e-169
Entrenamiento:Algoritmo	767.1671	6	4.052266e-01	8.749951e-01
Entrenamiento:Acelerador	2222.5828	9	7.826628e-01	6.326460e-01
Algoritmo:Acelerador	2626.0124	6	1.387090e+00	2.216297e-01
Entrenamiento:Algoritmo:Acelerador	254.5840	18	4.482474e-02	1.000000e+00
Residuals	60581.7752	192	NA	NA

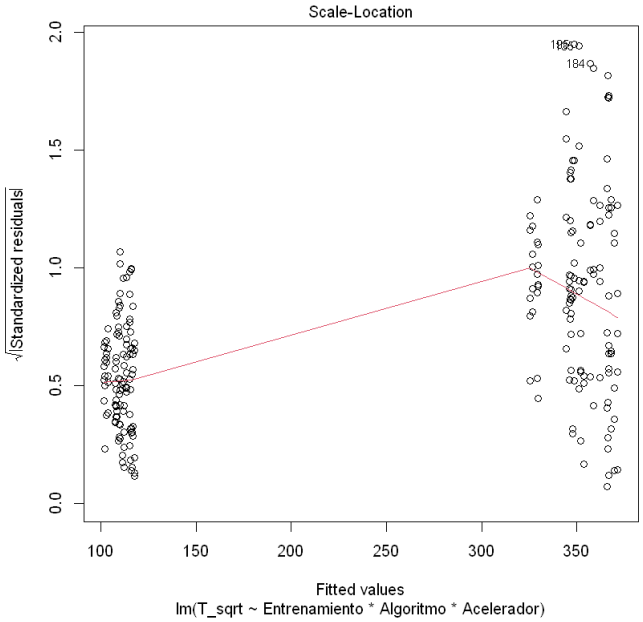





residuals(model)
fitted(model)
-60
-40
-20
0
20
40
60
100
150
200
250
300
350

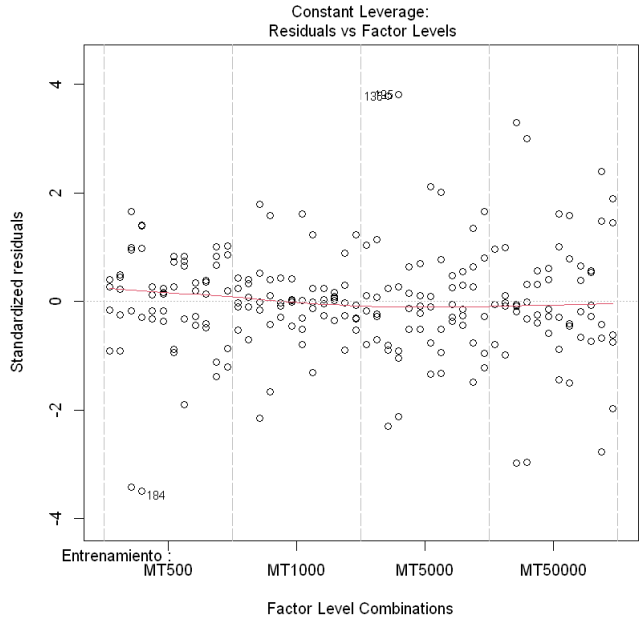

Residuals vs Fitted
Residuals
Fitted values
lm(T_sqrt ~ Entrenamiento * Algoritmo * Acelerador)
195
184
-60
-40
-20
0
20
40
60
100
150
200
250
300
350

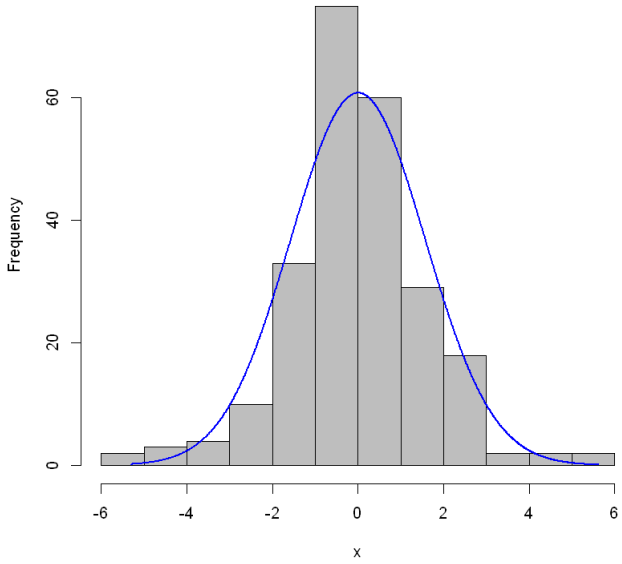

Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(T_sqrt ~ Entrenamiento * Algoritmo * Acelerador)
135
195
184
-2
0
2
4
-3
-2
-1
0
1
2
3

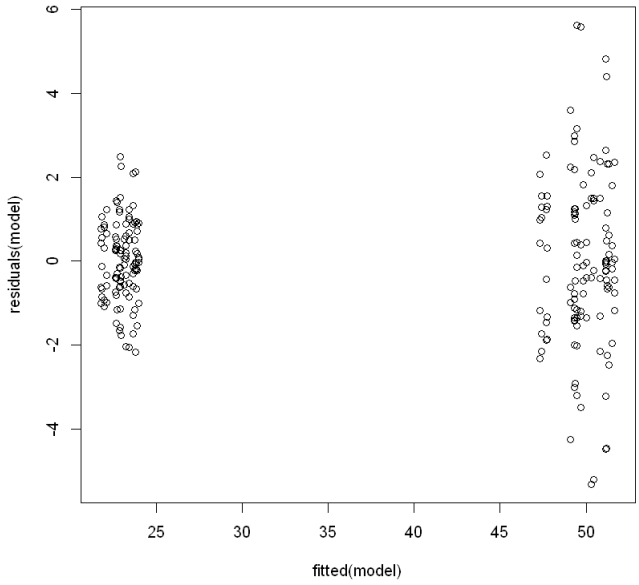


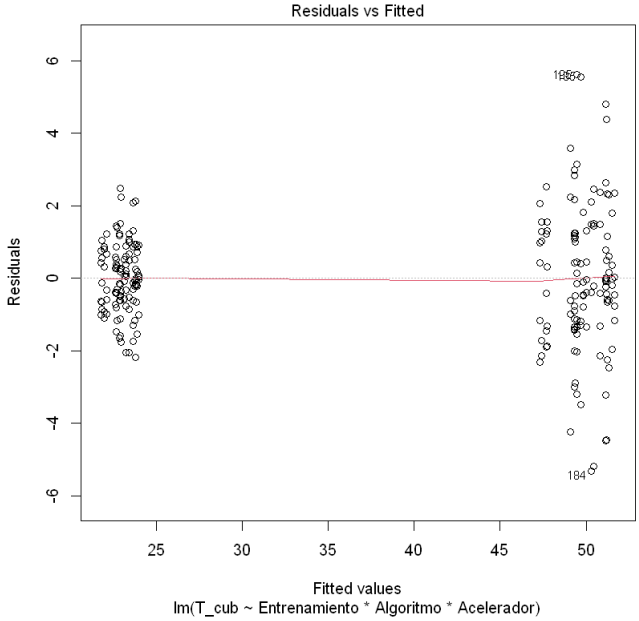
A anova: 8 × 4

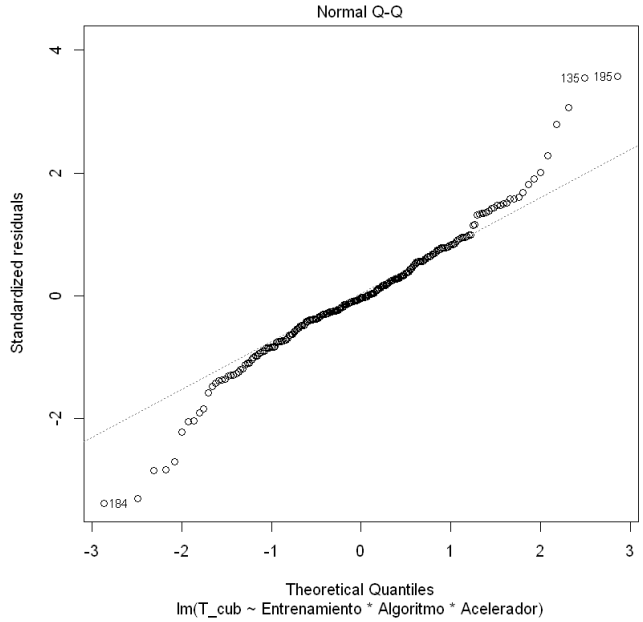
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Entrenamiento	108.368244	3	1.171012e+01	4.428384e-07
Algoritmo	106.700493	2	1.729485e+01	1.241257e-07
Acelerador	43031.289574	3	4.649899e+03	6.181819e-179
Entrenamiento:Algoritmo	8.821202	6	4.766032e-01	8.252271e-01
Entrenamiento:Acelerador	11.320782	9	4.077691e-01	9.299595e-01
Algoritmo:Acelerador	14.439634	6	7.801630e-01	5.864176e-01
Entrenamiento:Algoritmo:Acelerador	1.681881	18	3.029027e-02	1.000000e+00
Residuals	592.271492	192	NA	NA

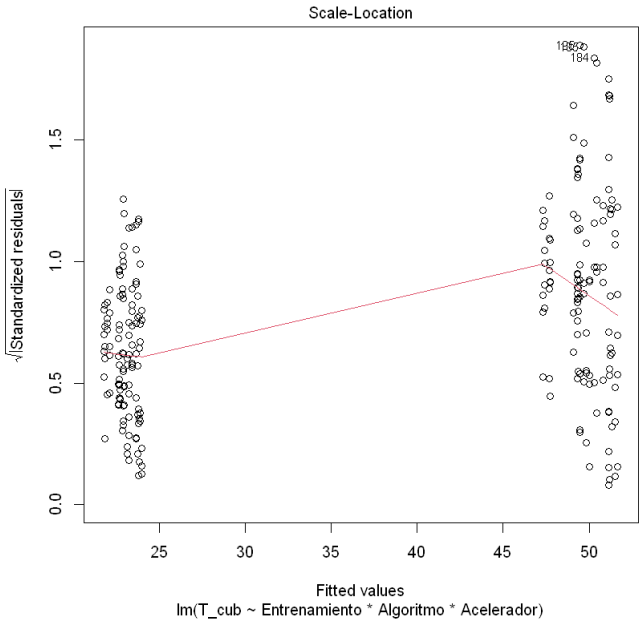



Frequency
0
20
40
60
-6
-4
-2
0
2
4
6
x


residuals(model)
-4
-2
0
2
4
6
25
30
35
40
45
50
fitted(model)

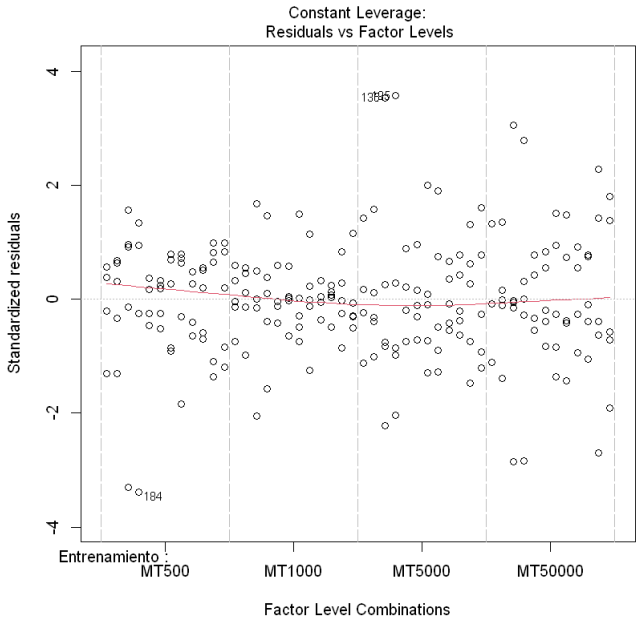

Residuals vs Fitted
Residuals
-6
-4
-2
0
2
4
6
25
30
35
40
45
50
Fitted values
lm(T_cub ~ Entrenamiento * Algoritmo * Acelerador)

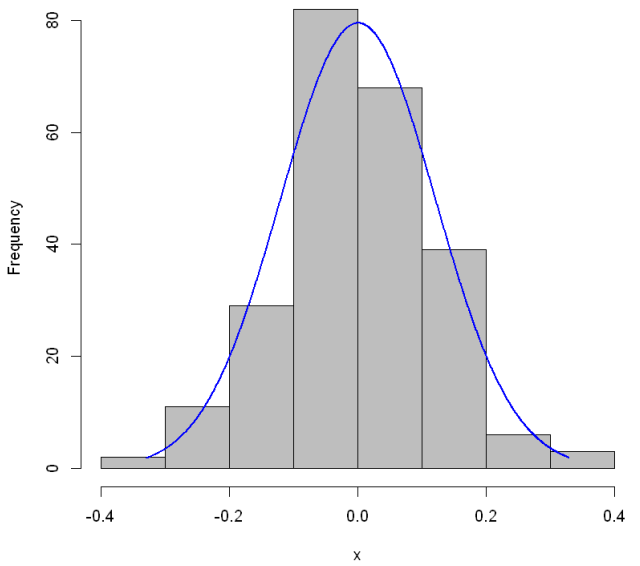


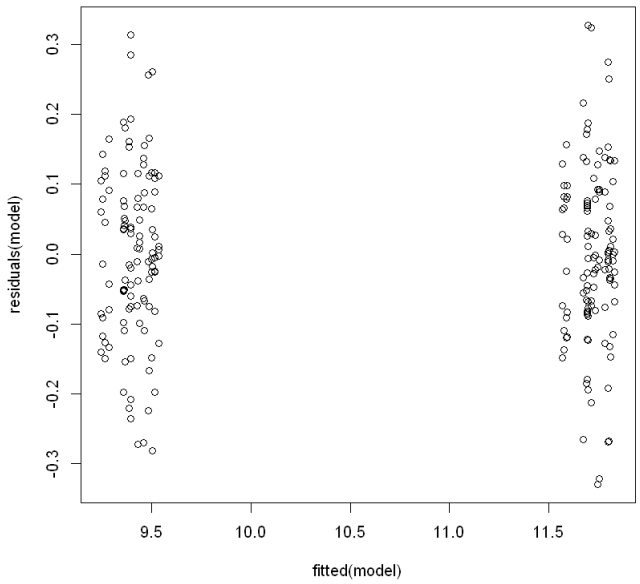


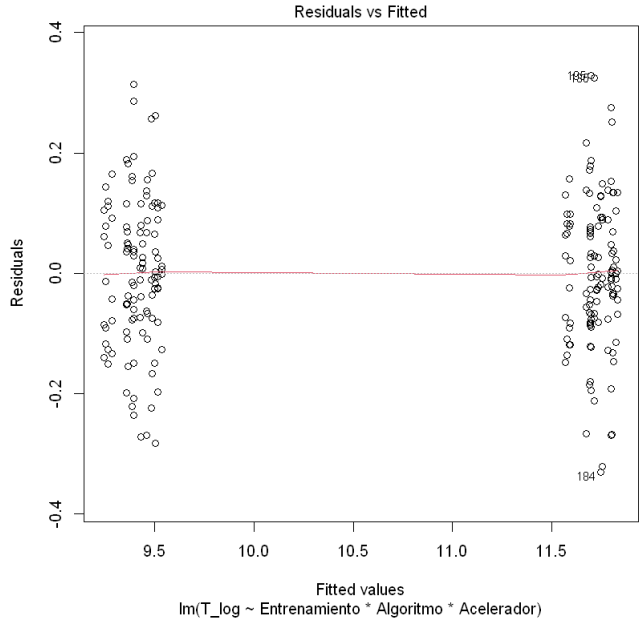
A anova: 8 × 4

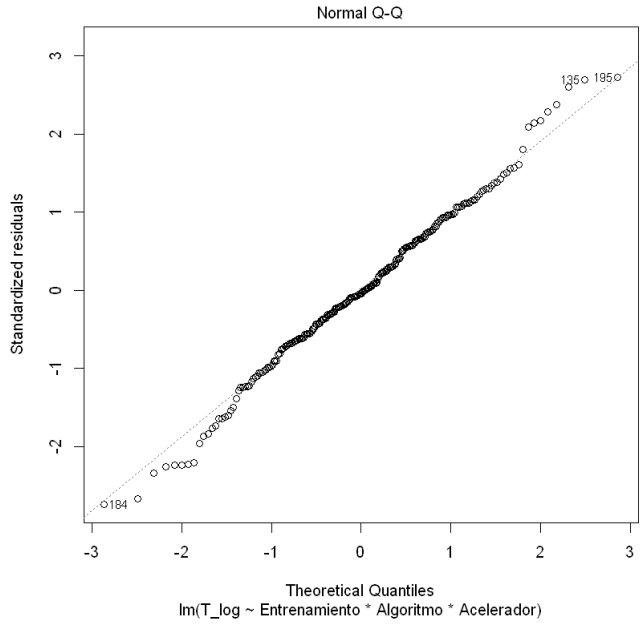
	Sum Sq	Df	F value	Pr(>F)
	<dbl>	<dbl>	<dbl>	<dbl>
Entrenamiento	7.831654e-01	3	1.445993e+01	1.565698e-08
Algoritmo	7.260307e-01	2	2.010754e+01	1.178790e-08
Acelerador	3.211197e+02	3	5.928976e+03	6.061320e-189
Entrenamiento:Algoritmo	6.655105e-02	6	6.143810e-01	7.186522e-01
Entrenamiento:Acelerador	5.762868e-03	9	3.546747e-02	9.999953e-01
Algoritmo:Acelerador	3.156218e-03	6	2.913733e-02	9.998927e-01
Entrenamiento:Algoritmo:Acelerador	7.031480e-03	18	2.163756e-02	1.000000e+00
Residuals	3.466308e+00	192	NA	NA

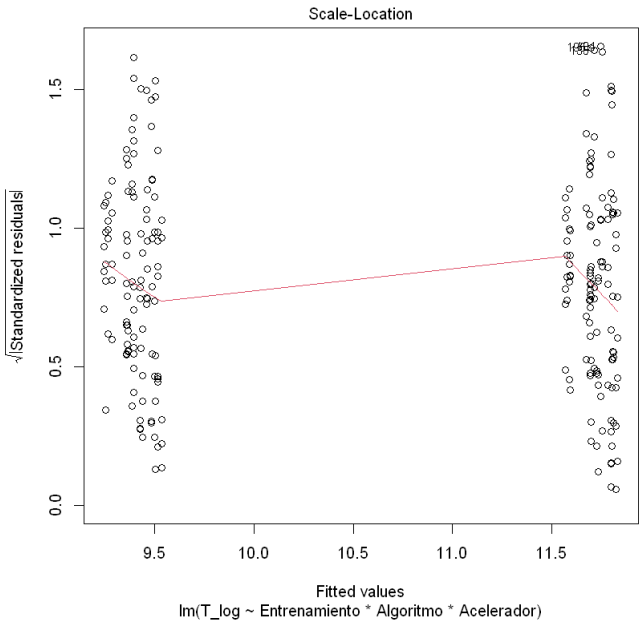

Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
Entrenamiento :
MT500
MT1000
MT5000
MT50000
Factor Level Combinations
184


Frequency
x


residuals(model)
fitted(model)


Residuals vs Fitted
Residuals
Fitted values
lm(T_log ~ Entrenamiento * Algoritmo * Acelerador)


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(T_log ~ Entrenamiento * Algoritmo * Acelerador)


Scale-Location
√|Standardized residuals|
Fitted values
lm(T_log ~ Entrenamiento * Algoritmo * Acelerador)

A anova: 2 × 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	47	0.4057217	0.9997739
	192	NA	NA

```
NOTE: Results may be misleading due to involvement in interactions

Loading required package: mvtnorm

Loading required package: survival

Loading required package: TH.data

Loading required package: MASS


Attaching package: 'TH.data'


The following object is masked from 'package:MASS':

    geyser


Note: adjust = "tukey" was changed to "sidak"
because "tukey" is only appropriate for one set of pairwise comparisons
```

A summary_emm: 3 × 7

	Algoritmo	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	Algoritmo C	10.48823	0.01502235	192	10.45205	10.52442	a
2	Algoritmo B	10.59413	0.01502235	192	10.55794	10.63031	b
1	Algoritmo A	10.61331	0.01502235	192	10.57713	10.64950	b

```
NOTE: Results may be misleading due to involvement in interactions
```

A summary_emm: 4 × 7

	Entrenamiento	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
3	MT5000	10.50388	0.01734632	192	10.46966	10.53809	a
1	MT500	10.51543	0.01734632	192	10.48122	10.54964	a
4	MT50000	10.60299	0.01734632	192	10.56878	10.63721	b
2	MT1000	10.63860	0.01734632	192	10.60438	10.67281	b

```
NOTE: Results may be misleading due to involvement in interactions
```

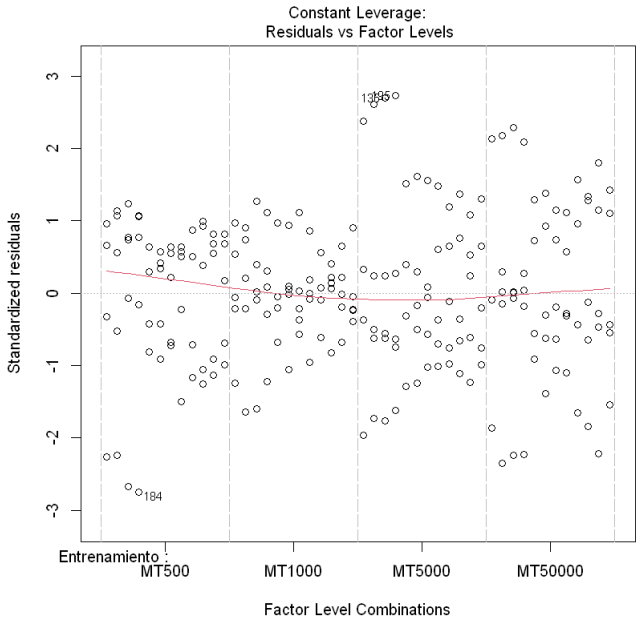
A summary_emm: 4 × 7

	Acelerador	lsmean	SE	df	lower.CL	upper.CL	.group
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
1	NA-NA	9.398931	0.01734632	192	9.364718	9.433145	a
2	NA-SW	9.418122	0.01734632	192	9.383908	9.452336	a
3	HW-NA	11.719681	0.01734632	192	11.685468	11.753895	b
4	HW-SW	11.724163	0.01734632	192	11.689950	11.758377	b

A data.frame: 12 × 11

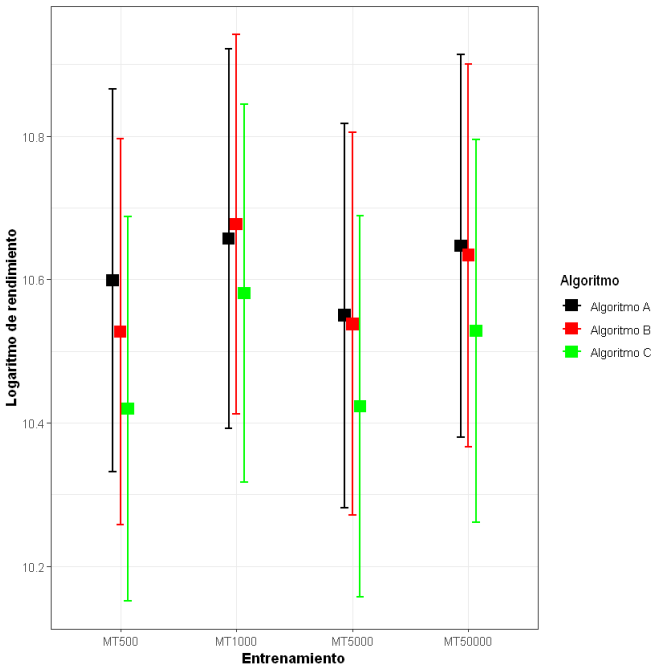
Entrenamiento	Algoritmo	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
MT500	Algoritmo A	20	10.599	1.195	9.159	9.523	10.508	11.843	11.905	0.267
MT1000	Algoritmo A	20	10.657	1.184	9.319	9.529	10.617	11.809	11.954	0.265
MT5000	Algoritmo A	20	10.550	1.200	9.159	9.351	10.603	11.641	12.037	0.268
MT50000	Algoritmo A	20	10.647	1.196	9.220	9.482	10.647	11.800	12.073	0.267
MT500	Algoritmo B	20	10.527	1.201	9.251	9.393	10.476	11.729	11.773	0.269
MT1000	Algoritmo B	20	10.677	1.185	9.406	9.526	10.678	11.818	11.968	0.265
MT5000	Algoritmo B	20	10.538	1.195	9.211	9.363	10.579	11.661	11.888	0.267
MT50000	Algoritmo B	20	10.634	1.195	9.321	9.442	10.654	11.771	11.941	0.267
MT500	Algoritmo C	20	10.420	1.200	9.105	9.306	10.411	11.622	11.693	0.268
MT1000	Algoritmo C	20	10.581	1.180	9.339	9.443	10.574	11.715	11.834	0.264
MT5000	Algoritmo C	20	10.423	1.191	9.133	9.240	10.434	11.574	11.748	0.266
MT50000	Algoritmo C	20	10.528	1.195	9.160	9.355	10.478	11.640	11.892	0.267

```
Warning message:
"Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead."
```



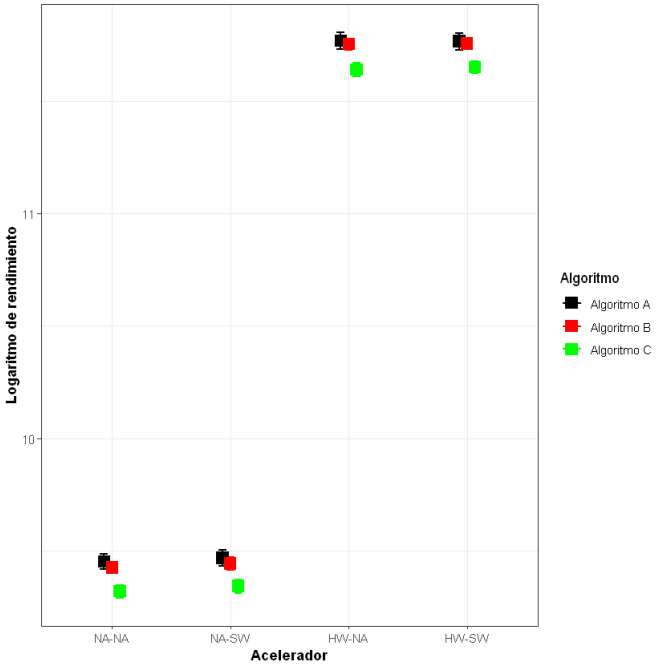
A data.frame: 12 × 11

Acelerador	Algoritmo	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
NA-NA	Algoritmo A	20	9.452	0.153	9.159	9.351	9.473	9.547	9.741	0.0342
NA-SW	Algoritmo A	20	9.468	0.166	9.184	9.329	9.497	9.590	9.764	0.0371
HW-NA	Algoritmo A	20	11.768	0.169	11.436	11.644	11.795	11.850	12.073	0.0378
HW-SW	Algoritmo A	20	11.766	0.167	11.419	11.655	11.794	11.856	12.053	0.0373
NA-NA	Algoritmo B	20	9.425	0.113	9.211	9.345	9.403	9.511	9.633	0.0253
NA-SW	Algoritmo B	20	9.443	0.123	9.244	9.363	9.421	9.540	9.653	0.0275
HW-NA	Algoritmo B	20	11.753	0.109	11.577	11.684	11.759	11.816	11.968	0.0244
HW-SW	Algoritmo B	20	11.756	0.115	11.517	11.673	11.769	11.830	11.941	0.0257
NA-NA	Algoritmo C	20	9.320	0.128	9.105	9.217	9.317	9.419	9.547	0.0286
NA-SW	Algoritmo C	20	9.343	0.136	9.114	9.232	9.373	9.451	9.547	0.0304
HW-NA	Algoritmo C	20	11.639	0.137	11.410	11.571	11.649	11.719	11.892	0.0306
HW-SW	Algoritmo C	20	11.651	0.118	11.472	11.553	11.674	11.703	11.863	0.0264



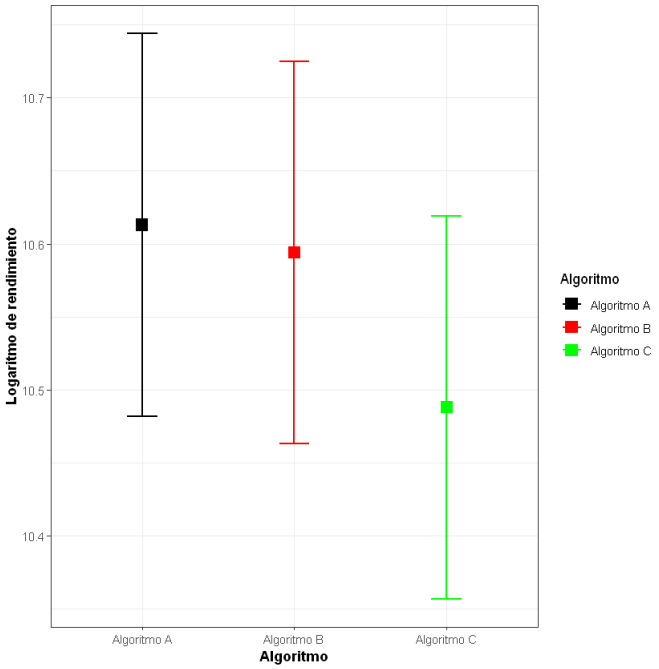
A data.frame: 3 × 10

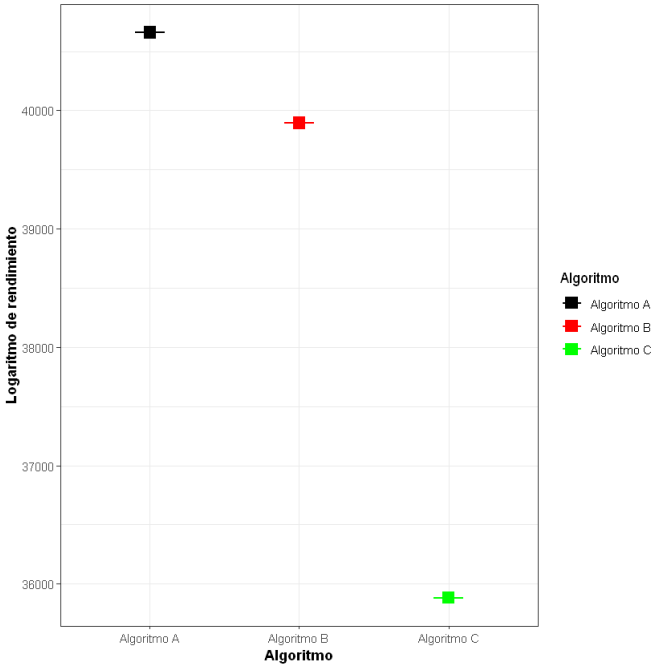
Algoritmo	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	80	10.613	1.172	9.159	9.482	10.591	11.792	12.073	0.131
Algoritmo B	80	10.594	1.173	9.211	9.413	10.585	11.766	11.968	0.131
Algoritmo C	80	10.488	1.171	9.105	9.348	10.478	11.656	11.892	0.131



A data.frame: 3 × 10

Algoritmo	n	mean	sd	min	Q1	median	Q3	max	se
<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Algoritmo A	80	40660.00	3.228443	9.159	9.482	10.591	11.792	12.073	0.361
Algoritmo B	80	39894.75	3.231673	9.211	9.413	10.585	11.766	11.968	0.361
Algoritmo C	80	35882.32	3.225216	9.105	9.348	10.478	11.656	11.892	0.361




40000
39000
38000
37000
36000
Logaritmo de rendimiento
Algoritmo A
Algoritmo B
Algoritmo C
Algoritmo
Algoritmo
Algoritmo A
Algoritmo B
Algoritmo C

In [ ]: